COLLABORATIVE SUPPLY CHAIN MANAGEMENT USING MULTI-AGENT APPROACH

Lokuketagodage Chandana Manoj Perera

(139106F)

Thesis submitted in partial fulfilment of the requirements for the degree of Master of Science in Artificial Intelligence

Department of Computational Mathematics

University of Moratuwa

Sri Lanka

December 2015

Declaration

I declare that this dissertation does not incorporate, without acknowledgment, any material previously submitted for a Degree or a Diploma in any University and to the best of my knowledge and belief, it does not contain any material previously published or written by another person or myself except where due reference is made in the text. I also hereby give consent for my dissertation, if accepted, to be made available for photocopying and for interlibrary loans, and for the title and summary to be made available to outside organizations.

Name of Student | Signature of Student
L.C. Manoj Perera | Date:

Supervised by
Prof. Asoka S Karunananda | Signature of Supervisor
Date:

Dedication

I dedicate this thesis to my family and friends. A special feeling of gratefulness for my loving wife, daughter and parents because their word of encouragement and push for immense achievement. I will always appreciate all they have done and their valuable thoughts. I dedicate this work and give many thanks to people at SLIDA for their help and especially for the library staff.

Acknowledgment

I would like to extend my sincere gratitude to many people around me those who have helped to bring this research success. I would like to thank Professor Asoka S Karunananda is my research supervisor and for the opportunity he has provided me for taking part of this Artificial Intelligence programme. I am so deeply appreciate his valuable guidance, help and professionalism throughout this course. I have no words to express my deep and earnest appreciation for my research supervisor.

I would also like to acknowledge Professor Priyan Dias for valuable thoughts. Many thanks to other lecturers, those who have deliver valuable knowledge and guidance. Many thanks for my best friends Tharindu and Dinesh for helping me to dominate the critical areas in technology. I will always appreciate Harendra contribution of many hours for proofreading throughout this document. Finally, I must express my gratitude to faculty, staff for the help and support throughout this master program.

Abstract

Supply chain management (SCM) manifests an inherently complex system. It involves a large number of distributed and interconnected entities which are far from the sequential execution while facing so much of uncertainty within the operational environment. The complexity in a supply chain management exercise to profit loss, customer dissatisfaction and sustainability of the business. Due to inherent complexity, in particular, dynamic nature of supply chain management, the traditional software technologies such as databases and web technology cannot be effectively used to model supply chain management. However, a large body of literature on research in a Multi Agent System technology (MAS) has demonstrated how the complex systems can be modeled to exploit the complexity as an opportunity to devise smart solutions which could not be achieved otherwise.

A research has been conducted to design and development of a MAS for the domain of SCM. A multi-agent solution for SCM primarily envisages the implementation of effective collaborations among the stakeholders within the process of supply chain domain. In this solution, each phase in the supply chain has been developed as an agent enabling communication, coordination and negotiation among the agents to achieve intended business goals. Our solution presents a decentralized, collaborative planning, architecture and agents are attached to different containers of the system. The identified stakeholders of a MAS solution for supply chain management include raw material supplier agents, manufacturer agents, transport agents, warehouse agents, retail supplier agents and customer agents. These agents activate when applicable and disappear from the deliberation if the interaction is counterproductive. This system has been tested using real world data, simulations and customer behaviors against agent response. Our solution has demonstrated the essential features of a MAS including, communication, coordination, negotiation, butter-fly effect, emergent feature and evolvebility. This system has elegantly demonstrated how initially active agents fail in the deliberation, and the final deal has gone to an agent who joined the deliberation in the latter part of the process. The final output of the system has achieved 80% of successes, in effective collaboration and agent interaction for delivering a smart solution.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Prolegomena

Increasing popularity and the penetration of Artificial Intelligence (AI) techniques turned into a wide spectrum of subject areas. The complex real world problems have made AI as a distinct technology in the 21st century. Over last six decades Artificial Intelligence techniques were showing the unique capacity to solve various complex real world problems which could not be solved otherwise. In particular the real world systems involved in large number of interconnected entities in a distributed environment under unpredictable uncertainty. With the increasing popularity in AI, numerous intelligence techniques including Artificial Neural Network [1] [2], Genetic Algorithm [3], and Expert Systems [4]. Among other AI techniques Multi Agent Systems (MAS) has provided effective solutions for problem solving where expensive communication, coordination and negotiations are required. Exciting applications of MAS technology have been reported in the complex real world problems such as Logistic Management [5], and Air Craft Maintenance [6]. Having recognized supply chain management as an inherently complex system, this project has been conducted to developed MAS solution for supply chain management. In this connection, this chapter presents aim and objectives, background and motivation, problem in brief, novel approach to supply chain management and structure of the overall thesis.

1.2 Aim and Objectives

The aim of this project is to develop Multi Agent System solution for supply chain management. In order to reach this aim the following objectives are identified.

1. To critically study the supply chain management domain with a view to identify current practices and issues.
2. Critically analyze the existing solution in supply chain management with the view to define the research problem and possible technology.

3. In depth study about MAS and its applications.
4. Design and implement MAS for supply chain management.
5. Evaluate the MAS using a real world scenario.

1.3 Background and Motivation

Identification of different entities in business process and how they work together with collaboration and sharing information to achieve intended goals for an organization success is an essential requirement of SCM [7]. These entities consists of material venders, suppliers, manufactories, distributors, retailers and customers and they work together as a network of agents in supply chain management. There are some developed legacy systems and client server architecture applications that are able to provide marginal benefits for the effective communication in SCM [1] [7]. Nowadays, centralized information processing systems, has replaced with decentralized complete software applications that ensure the independence in supply chain management [7]. Wan Sup Um and others [8] have proposed the mathematical model for effective communication among agents in supply chain environment, but Jorge E. Hernandez and others [7] has implemented decentralized collaborative planning architecture.

However, for last two decades there are numerous conventional software systems has been incorporated in finding improved solutions for solving complex system problems. On the other hand, multi-agent technology has emerged as novel approach for solving complex system problems, even though the entities are located in distributed nature. In addition to this, due to uncertainty of the decision making process, and different behaviors of the entities in supply chain management systems and their activities have been identified as rather complex in nature. Therefore, it is hard to find proper evidence of complete software application which has implemented for eliminating complexity in supply chain management. As a result of this MAS technology has been used throughout this project to eliminate the complexity and to offer effective communication mechanism for supply chain management process.

It is argued that the importance of effective communication architecture in supply chain while enabling the entities to communicate, coordinate and negotiate in operation. With the introduction of Multi Agent System (MAS) has accommodated these facilities needed to communicate in a distributed environment. Each agent in the network has different responsibilities according to the ontology they access and work together while planning and executing their tasks. In addition to this, inter organizational coordination and collaboration were not sufficiently achieved by traditional information systems and their successfulness depend on collaboration, information sharing and effective technology support. It is reviewed that, legacy systems, and current software applications that developed using client server architecture are lacking in effective communication and failed to perform accurately when they are in distributed nature.

However, some of the developed middleware components to communicate with legacy systems are not benefited as they were expected. In contrast, the systems that are developed using multi-agent technologies are achieved significant performance in communication with legacy systems in the domain of supply chain management. With available internet technologies and e-commerce systems are limiting the effective communication in different environment due to language translation, time zone variations and currency problems. Nevertheless, due to complexity in supply chain and power of multi-agent technologies has emerged prodigious concern in finding novel solution for SCM throughout this project.

1.4 Problem in Brief

Lack of proper coordination and communication in SCM environment have resulted in malfunctioning of whole supply chain management process leading to customer dissatisfaction, profit loss and increased wastage.

1.5 Novel Approach to SCM

Communication complexity is a predominant problem in supply chain management due to entities are operate in isolated environments and are lacking from information sharing and collaborative planning. Therefore, in this approach a multi-agent system has been introduced as an innovative technology to provide dynamic solutions for supply chain management. There are various requirements from the entities operate in supply chain process has taken as inputs to the system while providing customer satisfaction, increasing profit margin and reduce wastage are delivered as outputs from the system. Timely responses, information sharing, high performance under limited resources are identified as high-level features of the system. Yet another detail explanation about novel approach and comparison with conventional technologies are included in chapter seven.

1.6 Structure of the Thesis

Rest of the thesis is organized as follows. Chapter 2 critically reviews the domain of SCM by highlighting current solutions, practices, technologies, limitations defining the research problem. Chapter 3 describes the essentials of Multi Agent technology showing it is relevant to solve the SCM domain. Chapter 4 presents our novel approach to SCM with Multi Agent technology. Chapter 5 is on the design of MAS for SCM. Chapter 6 comprises details of implementation of the MAS solution for SCM. Chapter 7 illustrates a real world problem using novel approach. Chapter 8 reports on evaluation of the new solution by explain evaluation strategy, participants, data collection, representation and analysis. Chapter 9 concludes the outcome of the research with note on further work.

1.7 Summary

This chapter describes the full picture of the whole research project showing research problem objectives, hypothesis and novel solution. Next chapter will be on literature review of SCM practices, technologies and issues with a view to define the research problem.

Chapter 2

Current Trends and Issues in SCM

2.1 Introduction

There are legacy systems have been developed for supply chain management [7]. Such systems include advanced planning systems (APS), enterprise resources planning (ERP) and e-commerce systems, and they have failed to achieve benefits of collaboration among the entities. Supply chain can be viewed as connected ultimate suppliers and ultimate customers. The suppliers distribute merchandise at the right quantities to the right locations at the right time to minimize the system wise cost, while satisfying the service level requirements. In supply chain management performance depends on willingness and coordination with others. As a result of this one entity in supply chain is necessarily affecting the performance of the other entity. However, suppliers are inclined to make own decisions in SCM. They involve in many organizations but, sometimes have to depend on decisions made by manufacturers [9]. The retailers are also equally powerful by compared to suppliers although they have conflicting requirements. Therefore, usefulness of decentralized decision-making mechanism is an apparent requirement in all conditions.

Simulation in supply chain using a multi agent approach have been introduced by Swami Nathan and others to understand supply chain dynamics [10]. According to this solution it is evident that their theory has focused on simulation rather than automation of supply chain functions. The methodology is to predict future trends in supply chain using theory of probabilistic approach. We realized that with the considerate of the above theory and their limited consideration about the agents and use of ontology for communication. Nevertheless, using JADE [11] like framework with message passing standards (Foundation for Intelligent Physical Agents - FIPA) [12] and with access to different ontologies for effective communication has made enormous impact for successfulness of SCM decisions.

Effective communication and information sharing improve working capital by 25% [13]. Regardless of the business type in SCM process, primary objective of the every business entity is to earn maximum profits by increasing working capital, streamlining accounts, managing debtors and eliminating unnecessary cost. Therefore, the importance of accurate communication mechanism is an essential factor for the SCM. Among others, introduction of agent technology and JADE platform has fulfilled the needs of effective communication mechanism. SCM is regarded as a strategic essential factor in obtaining great deal of competitiveness in the business environment [8]. In SCM if there is one single company exists in one directional way collaboration is may not be useful and not complicated. However, cumulative business growth and connection with other companies in the business process an agent technology can be introduced as promising candidate in supply chain management. SCM, network has facilitated different functions of distribution options for procumbent materials, convert it to intermediate and finished goods and finally distribute over the end customers [14].

According to the swarm intelligence collections of agents work like ants or honey bees [15]. Their searching starts from parallel points and each member carries out intended operation to maximize utility function for sharing information with others. Individual functions of these entities are quite simple but their collective results are highly benefited. They access local knowledge and produced outstanding global results without any interaction from centralized authority. The introduced gravitational search algorithm is practical to swarm optimization algorithm, which inherits the features of Newtonian laws of gravity. M. Ojha [10] also stated that swarm intelligence techniques in supply chain has provided optimized solutions in non-leaner environment for a dynamic problem domain. However, he has introduced mathematical model for to resolve anomalies in supply chain. But the little consideration has made the distributed architecture and the identification of entities involve in SCM because all agents has different roles and responsibilities among the entire process.

As a new technology paradigm for conceptualizing, designing and implementing computer applications and to eliminate potential issues related to current software applications was achieved using agent technology [16]. According to this solution have introduced by X. Li and others described how to apply agent mechanism to wrap the existing systems. Their solution manifest legacy applications, enterprise resources planning systems (ERP) and advanced planning systems (APS) are converted into agents to maintain the centralized coordination and to accomplish organizational objectives. In addition to this, it is evident that the introduced framework have effectively benefited for centralized coordination in supply chain, but they have not considered the specific technology for implement agents in real environment.

Supply chain management is a solution to the some industrial issues [17]. Therefore the bullwhip effect [18] can be identified as a major problem in the domain of SCM. In SCM a single modification has done by the end customer and it affects to the entire process is named as bullwhip effect. From the industry viewpoint, production planning, inventory management and transport are the main areas of finding significant problems in SCM. However, handling these entities as an individual and to manage them to behave using their own decisions is rather complicated and difficult. For instance, transport and the inventory management are known as N-P hard problems and are very difficult to handle because all of these entities are interdependent. In order to eliminate above constraints needs a system, which supports collaborative planning and information sharing. Object modelling and simulation are new mechanism have developed for supply chain management [19]. Object modelling have facilitated modelling and individual agents. However, throughout their implementation have not indicated use of ontologies, message passing techniques and collaboration for effective communication.

Procedural Reasoning System architecture (PRS) is one of the best architecture of multi agent technology [20]. It has been deployed in many industry applications not only in supply chain management but also in other complex commutation systems. This architecture can derives a solution to resolve communication issues arrived in distributed multi-agent reasoning systems. However, as a simplest architecture for an agent can be introduced as belief, desire and intention architecture also called as BDI architecture [21] which consists of simplest characteristics, associated for intelligence agents. Supply chain is denoted as complex dynamic system and every member within the system engage the local interactions [22]. MAS have permitted satisfactory results for a problems with higher complexity by using classical mathematical model

[23]. Development of software agents for distributed data processing can be accomplished by using a framework [24]. They also discussed that growth of contact centers, e-commerce and value chain has resides additional problems of entropies data management and exploitation. However, it has included limited information about the successfulness of the solution in real time and acceptance of the end customer.

Layered multi agent system architecture is based on agent commitments [25]. It has illustrated that agent are as components and their interconnections specified as a term of commitments. Combine architecture is an apparent solution for eliminate communication issues in SCM [26]. Today's most of the supply chain functions are static and relay on long term relationships among trading partners, but more dynamic and flexible practices between entities in supply chain management provide better solution for customers and suppliers [27]. With the introduction of theoretical framework model for collaborative inventory management have effectively benefited successfulness of the SCM [28]. Within their framework have synchronized the customer trends with inventory decisions throughout the SCM network. However, the intended results were limited to conceptual and theoretical implementation. To optimize supply chain performance need effective coordination among the entities but due to dynamics of the business operation in an organizational environments are facing difficulties in all aspects. Therefore, sometimes it leads to delayed shipments, production failures, employees are informed as sick and customers are change their requirements [29].

In real world business environments there are no obligations for any organizations to remain with the supply chain. According to their own judgment, companies can join or leave at any time from the SCM process [30]. In particular, our solutions have implemented using JADE environment and included supply chain functions in real time with more customer involvements for the purpose of wastage reduction, increase overall profits margins and finally to satisfy the customer requirements. If the designed system includes high-level management capability and able to control the complex and dynamic environment then it comprises high commercial importance [31]. Those systems consist of air traffic control systems, telecommunication networks, space vehicles, business processors and medical services. The development of conventional solution for this systems have increased and shown that very expensive and difficult to maintain over the time. However, agent oriented systems offered radically different view for computational solution with qualitative change. MAS technology has inbuilt mechanism to communicate with ontologies. This MAS framework consist of connectors to communicate XML files and MYSQL databases [31]. A Multi-agent engine, Virtual world, Ontology and Interfere are four components provided by the multi-agent systems while enabling advanced communication standards for sending and receiving messages among active agents [32].

Every agent has decision making authority and living in some kind of an environment. They act autonomously based on their observation skills while accessing domain knowledge have defined in ontologies. In multi-agent applications collaborative decision have committed successful results rather than an individual decision making process. Each agent starts actions individually, but the outcome of actions are consists of combinations [33].

Regardless of the application domain, planning is not an individual activity, but need collaboration of others. When an agent start to plan abut certain task it is depend on the situation of the environment. As an example agent has to plan route from one place to another. There are two types of agents in the example including planning agent and reactive agent. However, successful results have received when agents are worked together rather than individuals [34]. When measuring supply chain performance, time based performance have received substantial outcomes. To address the performance in SCM, there are several frameworks have proposed by different researchers [35]. Beamon and others have proposed three types of performance measures in SCM. The resource measure, output measure and flexibility measure have been identified as considerable techniques to measure the supply chain performance [36].

A business community and academia are very extensive on literature regarding supply chain collaboration, but they are not always rely on the targets [37]. There are few writers especially focused on the cultural aspect of the collaboration which has identified as a serious mistake. According to the recent survey have conducted by Supply Chain Management and Computer Science Corporation revealed that the collaboration is most serious problem in complex communication. However, there is no specified mechanism to understand the essentially of collaboration. R.P. Kampstra and others mentioned that the leader have strong command during collaboration. According to the multi-agent approach, agents have assigned to complete the tasks with equal powers. Therefore, in our solution collaboration and equal power among agents have achieved successful results.

In multi-agent applications there is no proper evidence of centralized control [38]. The most prominent feature of complex system is decentralized decision making rather than centralized environment. Every complex systems consist of global and/or local ideologies, rules, laws and algorithms for agent to follow. George Rzevski [32] have implied that an agent operations have committed far from the equilibrium. Since early 1980s SCM has become most popular concept among management in logistic operation [39]. The real concept in SCM is based on integration of components. Therefore, integration can be introduced as strategic and operational. However, throughout this research paper have taken important strategies to convey the message of integration is best factor for supply chain management. In general, our solution also followed the importance of integration and collaboration.

The logistic management is sub instance of SCM. It also consist of physical distribution of material and material management for different geographical locations [40]. The field of business logistic have various categories including planning, implementing, identify the cost effective flows, raw materials, inventory, finish goods, storage facilities and customer information. Logistic management is created value for suppliers as well as customers in business perspective. Their operations are based on time and the place, sometimes product have no value or limited value due to products are not within the required time and place requirements.

According to the Ronald H. Ballou, the logistic management under supply chain include various value adding processes for numerous of reasons. The costs are significant and globalization of industries are some of the features have identified by Ronald H. Ballou throughout his research paper. However, limited evidence have found during his research content about solution rather than the problem identification. An innovative research projects have conducted during last decade to find the solutions to improve transport logistics [41]. With regard to the level of adoption is has not achieved the satisfactory level due to limitation in implemented methodologies. The European Union (EU) have funded to investigate the reason behind of the slow integration of advanced technologies, to improve the outcome of logistic management.

However, Thorsten Blecker and others have identified barriers to successful integration in logistics management including, financial matters, missing/limited hard facts, misaligned performance metrics, limited contract durations, lack of proper legal/institutional framework, customer security problems and lack of trust. Throughout their content have not taken an important position to implement continuous negotiation mechanism to avoid the identified barriers in logistic management. The process of SCM can be viewed as emergent field of practice as well as emerging academic domain [42]. The progress of SCM is ultimately depends on each other collaboration. However, during this research content John Storey have illustrated the limitation of people/entity behaviors and dimensions. In most instance of the research findings during this chapter have mentioned that, SCM as a complex problem in nature, but solutions are not adequate to control distributed communication complexity. Therefore, most of the researchers concerned on finding limitation in the process rather than new technology adoption.

In view of the above, we identify the research problem as the lack of proper coordination and communications in supply chain environment have resulted in malfunctioning of whole process and leading to customer dissatisfaction, profit loss increased wastage and sustainability of the business process. A new multi-agent system technology have introduced, implemented and evaluated in upcoming chapters in this research project as a novel approach to eliminate supply chain complexity.

2.2 Summary

In this chapter we have identified the research problem especially in Supply Chain Management. SCM is continuous process, but some activities are operate in parallel. To address the complexity in parallel communication have taken important position among researchers to find a smart solution. In general, increased complexity in distributed environment have resulted unexpected outcomes.

According to views of the researches have stated in this chapter, conventional software technologies are far away when compared to the multi-agent system technology. However, their solutions have been included smart technology such as object oriented architecture and multi-agent techniques, but limited consideration have taken to construction of ontologies to represent knowledgebase. Therefore, during this research project important strategies have taken to develop not only MAS application but also to construct ontologies to represent knowledge base. The next chapter includes essentials of multi-agent technology and also described different features of an agents in multi-agent applications.

Chapter 3

Essentials of Multi Agent technology

3.1 Introduction

Previous chapter described a large volume of literature review about supply chain management in multi-agent technology and their successfulness and some weaknesses. In this chapter illustrates that key technology challenge in automation process of SCM using multi-agent technology and the importance of it. The concept of multi-agent systems has become imperative in Artificial Intelligence and computer science due to power of its features and wide industry acceptability. The theory behind agent is, question about what is an agent and the use of mathematical formalisms for representing and reasoning the properties of an agents [18].

However, multi-agent system technology is combination of group of agents sharing information and acting collaboratively to achieve common goals. In other words MAS can be define as collection of heterogeneous computational entities with their own problem solving strategy and interaction with others to achieve intended goals [43]. The communication is one of the key components of multi-agent system. In fact, agent need to communicate with other agents, system resources and some legacy systems they need to negotiate, corporate and collaborate and so on [44]. The benefit of the MAS is, it can be used to solve problems which are very difficult in nature and impossible to solve using object oriented techniques or rather conventional systems.

There are many software projects have been used multi-agent technology to eliminate communication complexity in various domains including supply chain complexity in distributed environment. In the past few years multi-agent technology has emerged and combine with distributed artificial intelligence (DAI) as new research topic to provide new approach for software engineering [45]. Nevertheless, an agents technologies have shown new paradigm for solving old and new problems in any complex environments [44]. Throughout this chapter has illustrated that the features of agents, behaviors, architecture, industry acceptability and comparison with conventional software applications.

3.2 Agent Features and Behaviors

Possibly the most general way that the term of agent has been used to denote hardware or software components, this includes the following properties.

3.2.1 Autonomy

Agents are operate without the direct involvement of humans and others. They retain some kind of control of their actions and internal state. Also consist of independence and autonomy while enabling knowledge or representation capability for others.

3.2.2 Social Ability

Agents interact with other agent (Sometimes humans) via some kind of agent communication language. Agent Communication Language (ACL) has proposed by the Foundation for Intelligent Physical Agents (FIPA) as a proposed standard language for agent communications. The core fact about the FIPA is to provide interoperability among agents. Knowledge Query and Manipulation Language (KQML) is another proposed standard which offers protocol for exchange information and knowledge [46]. Agents are communicating with other agents via message passing techniques. It is an expensive procedure and it has achieved using ACL message system.

3.2.3 Reactivity

Agents can perceive the environment changes and can be response timely manner what has occurred on it. It may be from physical world, change in the graphical user interface, collection of other agents, from internet and sometimes combination of all. [18].

3.2.4 Pro-activeness

They are not simply act in response to the environment, but able to exhibit the goal directed behavior by taking the initiatives.

3.2.5 Difference Behaviors of Agents

An agents can execute different behaviors concurrently while communicating with other agents. Java Agent Development Environment have been introduced a mechanism for computerizing the agent behaviors [44]. There are three primary types of behaviors available with JADE with diverse features. The one shot behavior is designed to provide single execution and it is execute only once. The cyclic behavior is programmed to provide never complete route and it is operations is called until stop. The final behavior is generic behavior and it is designed to provide continues tasks until specific condition is met.

3.3 Multi Agent Architecture

Agent architecture consists of software engineering models of agents and they primarily concern with designing software and hardware to satisfy the properties provided by the theories [45]. To develop software model for an intelligence agent highlights the essential of agent architecture.

As a simplest architecture for an agents can be introduce as belief, desire and intention architecture also called as BDI architecture which consists of simplest characteristics associated with intelligence agents [44][47]. Belief best describe the information state of agents, desire illustrate motivational state of agents and intention represent the deliberate state of agents. The architectures can be divide into four main categories respectively logic based, reactive, BDI and layered according to the time they have introduced. A complete description of the features in multi-agent technology against other systems have depicted in Appendix A.

3.4 JADE as Development Environment

When developing multi-agent systems the environment and architecture is an immense advantage for the successfulness of the solution. Java Agent Development Environment (JADE) is a middleware component that facilitates the development of multi-agent applications. It consists of runtime environment, library of classes and graphical tools.

3.4.1 A Runtime Environment

A runtime environment describes the place where agent can "live" and must be activated in a given host or terminal. One host can contain one or more agents and that running instance is called as container. A set of activated instance of containers is denoted as platform [48]. In a JADE platform main container should be activated in runtime which is the first container and has the administration facility over the other containers. The other containers should be registered with main container and need to configure with name of the host and port number. In a network some other platform is activated the normal containers can be registered with main container. The advantage of this platform architecture is the registered agents under any containers can be communicate via message passing. Figure 3.1 illustrates the above runtime environment with simple scenario showing JADE platform with multiple containers.

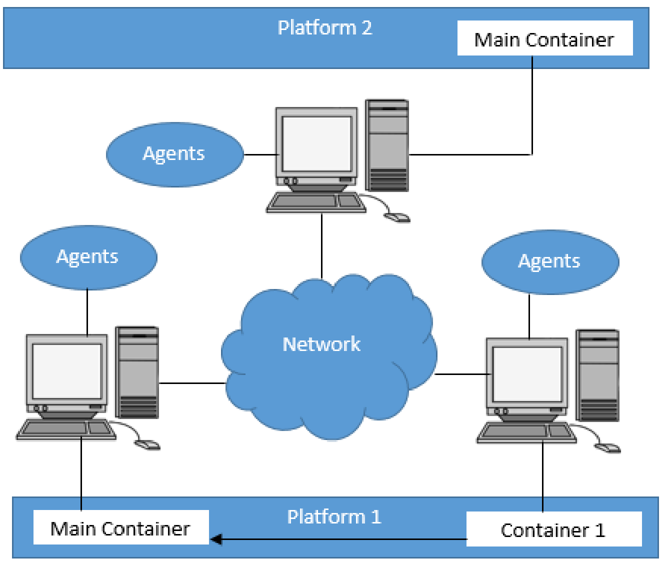


Figure 3.1: Containers, Platforms and Agents

The Figure 3.1 describes the arrangement of platforms, containers and agents in a networks environment. An agent can communicates with the other agents regardless of the platform and container that they have attached or authenticated.

3.4.2 Library of Classes

The programmers are independent to use library of classes to implement agent behaviors. As described in the above paragraphs in this chapter, to implement different behaviors of an agent the programmers have to incorporate the intended java classes which are included in JADE framework.

3.4.3 Graphical Tools

The crated agents should be authenticated, monitored and administrated when they are in runtime. Therefore, set of graphical tools are available under JADE environment to simple agent administration such as agent migration, stop, resume, suspend and termination whenever necessarily.

3.5 Industry Acceptability and Usage

A multi-agent systems are extremely popular in wide variety of industry applications, ranging from complex systems, mission critical systems and small systems for personal assistance [49]. Industry applications have taken early advancement in multi-agent system because they are the first multi-agent systems were demonstrated and tested of their initial potential. No conventional system have shown feasible solution for solving problems related Distributed Artificial Intelligence (DAI) [43]. Therefore, MAS is seen as major research space for developing industry application especially in the areas of distributed computing and distributed artificial intelligence.

3.6 Comparison with Conventional Technologies

Conventional software application have shown limited potential for solving inherently large and complex problems. One of the major drawback of conventional systems are fail to establish connection with modern applications. This issue is still remain unsolved due to communication difficulties with conventional system when sharing information with modern software applications. However, with the introduction of multi-agent systems have shown strong capacity for establishing connection with conventional application especially in information sharing.

Today's organizations are diversified their business functions and are not operate in single location. The operations are rather complex and requirements are different when operate in distributed environment. Therefore, conventional systems are lacking to provide essential characteristics when the business functions are perform in distributed nature. However, the problem domain is complex, unpredictable and large, then the only one way is, it can be addressed by modularizing the components [50]. This decomposition or modularization can be developed by replacing agent technologies because, agents are permit to use most appropriate paradigm to solve appropriate problem which could not be solve by using conventional technologies. A detail description is explained using graphical notations in chapter seven.

3.7 Summary

This chapter discusses the important and advantages of multi-agent technology to solve communication problems arise in distributed nature. It also described the features of agents and their different behaviors. The next chapter will discuss approach to SCM using multi-agent technology.

Chapter 4

Approach to SCM with Multi Agent Technology

4.1 Introduction

In the previous two chapters we define the research problem as the inefficiency in SCM and the existing solutions by describing why MAS technology could be a potential technology to develop novel solution for SCM. This chapter presents our approach by describing the hypothesis, inputs, outputs, process, features and users for novel solution for MAS based SCM. The new solution has been named as MASSCM and is the acronym for automated Multi Agent System for Supply Chain Management.

4.2 Hypothesis

MAS will be able to model complex system of SCM to address the issues of communication among the parties involved in the process.

4.3 Inputs to the System

MAS for SCM have been design to accept multiple inputs coming from different entities of the supply chain process. The table 4.1 shows that the inputs from corresponding agents.

Inputs	Agents
Raw material quantities	Raw material supplier
Transport locations	Transport
Delivery dates	Transport/ Manufacturer
Dates and holidays	Human Agent
Manufacturer details	Manufacturer
Minimum order quantities	Warehouse/ Retail
Reorder levels	Warehouse/ Retail
Reorder quantities	Warehouse/ Retail
Demand	Customer

Table 4.1: Inputs by Various Agents

4.4 Outputs to the System

There are two modes of output generated by the system. These outputs are coming as organizational related and the customer related aspects.

Organization	Customer
Reduced wastage	Receive products in right time
Increase profits	Receive right quantities
Final customer satisfaction	Receive products in right quality

Table 4.2: Outputs of the System

4.5 Process of SCM

Having entered the inputs to the system uses MAS technology and ontology to generate the outputs. In SCM process agents are perceive the environment changes and can be response timely manner what has occurred on it. It may be from physical world, change in the graphical user interface, collection of other agents, from Internet and sometimes combination of all. They are not simply act in response to the environment but, able to exhibit the goal directed behavior by taking the initiatives. In this process seven major types of agents are defined to the system. In particular, the agent categories, alias names and no of active agents with their assigned tasks are explained in chapter six. The knowledge required these agents to operate are stored in a common domain ontology and personal ontology.

Agents are autonomous and work together with collaboration and information sharing to complete given tasks. Therefore agents are basically access their personal ontologies to generate decision about raw material quantities, who are the selected suppliers, pricing, discounts, quality of the products, after service, and the past records of entities in SCM. However agents are access common domain ontology which has broad description of the entire supply chain process and made available to any agent with define permissions. Inventory management rules are introduced such as FIFO (First in First Out), LIFO (Last in First Out) according to the product category and availability of stocks.

As a promising technology Java Agent Development Framework (JADE) have been used to automate supply chain management process. The solution is used 90 % of JAVA technology while some reporting mechanism and ontology management are automated using Hyper Text Pre Processor (PHP), JQuery, and Java Scripts. In addition, NetBeans IDE is used as development environments.

4.6 Features of SCM

Overall features of the proposed system are not derive from the inputs and outputs and are basically non-functional requirement of the system. Agents are created using Java Agent Development Framework (JADE) therefore each agent in the framework consumes limited resources and produce high performance. Despite of other multi agent technologies JADE is available as free software component hence development cost is marginal. Ability to perform under limited resource environment, installation and access through the mobile devises are increased the rapid growth of multi-agent technology. Instead of the standalone environment the agent are accessible through the web interface with minimum bandwidth.

4.7 Non Functional Requirements of the System

Non-functional requirement of the system is used basically to identify the system operation in all conditions. In other words, it described the qualities/constraints of the system. No of concurrent active agents in a given moment and number of database connections activated are and some of the non-functional requirements for SCM. The system have maintained the proper security model to accommodate the non-functional requirement while encrypting the user credential for accesses and modifications to the knowledge base. MYSQL server have configured to provide necessary concurrent database connections without affecting to the performance of the system. However, the system architecture have designed to accommodate new requirements will arise in the future to meet system extensibility.

4.8 Users of the System

The developed application basically aims to satisfy the requirements of the stakeholders how has interested to the system. In SCM has wide spectrum of users interested in different functions of the system. Users of the system include raw material venders, manufacturers, transport agents, warehouse agents, insurance companies, retailers and customers.

4.9 Summary

This chapter discussed the overall picture about the novel approach to the supply chain management and presented that hypothesis, inputs, outputs, process, users and finally what are the non-functional requirements are been achieved by the system. Next chapter will illustrates the major design architecture of the SCM and who are the entities involve in the process and their relationships.

Design of MAS for SCM

5.1 Introduction

Previous chapter described the approach to the supply chain management including hypothesis, inputs, outputs, process and features of the system. In this chapter the main objective is to presents a high-level understanding of the design architecture, modules and relationships among the different entities are involved in supply chain management.

5.2 Two Components Architecture of SCM

In the context of the SCM, information sharing and collaborative planning are the most principal features have been used for to eliminate complexity in distributed communication. In addition to that it can be introduced as parallel components and are highly coupled each other in communication. The Figure 5.1 illustrates that simple architecture of SCM.

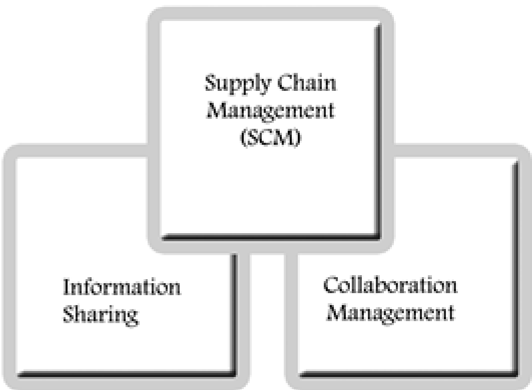


Figure 5.1: Two Components Architecture

Figure 5.1 the two components architecture of the SCM is presented as fundamental understanding of information sharing and collaboration to achieve projected business goals. Each agent in supply chain process have to complete their assigned tasks while enabling common features such as coordination, communication and negotiation.

5.3 Essential Entities in SCM

There are six main categories of entities (modules) have been identified namely raw material suppliers, manufacturers, transport agents, warehouse agents, retailers and customers. All of these entities are access common domain ontology and their personal ontologies while information sharing and collaborative planning. They sometimes found as hierarchical within the supply chain process, but in most cases they have a parallel behavior. Therefore, Figure 5.2 illustrated the different entities (modules) are operated in the process and their contribution in number of agents.

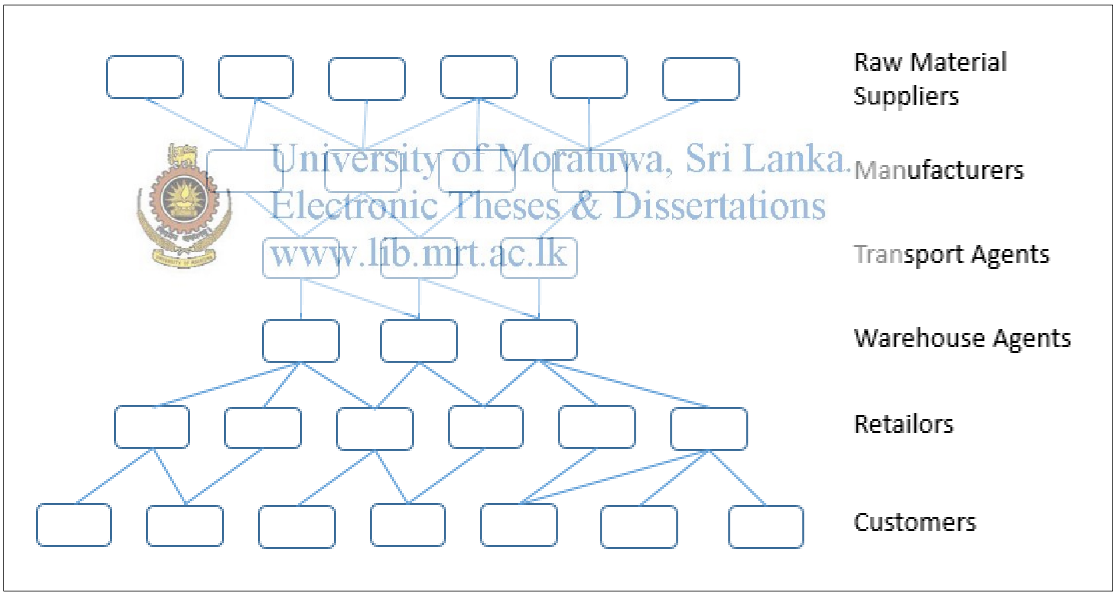


Figure 5.2: Module Based Architecture of SCM

Figure 5.2 illustrates that, who are the responsible entities are involve in supply chain management process and their dependencies. There are limited number of manufacturer agents are operate in the process and they hold significant responsibility by compared to raw material suppliers. In other words, they have more control over the entire supply chain process. However, retail supplier agents are dealing with the customers and any bullwhip effect from the customers end will leads to substantial change in whole supply chain management process. Therefore, handling the behaviors of each entity (module) group is found as demanding tasks and essential to implement comprehensive ontology including common domain and personal to smooth functioning of the SCM process.

5.4 Relationship between Entities in SCM

In this chapter a main purpose is to illustrate relationship among the entities (agents) are associated in SCM process. The Figure 5.3 describes the overall architecture of SCM and relationship among each agent.

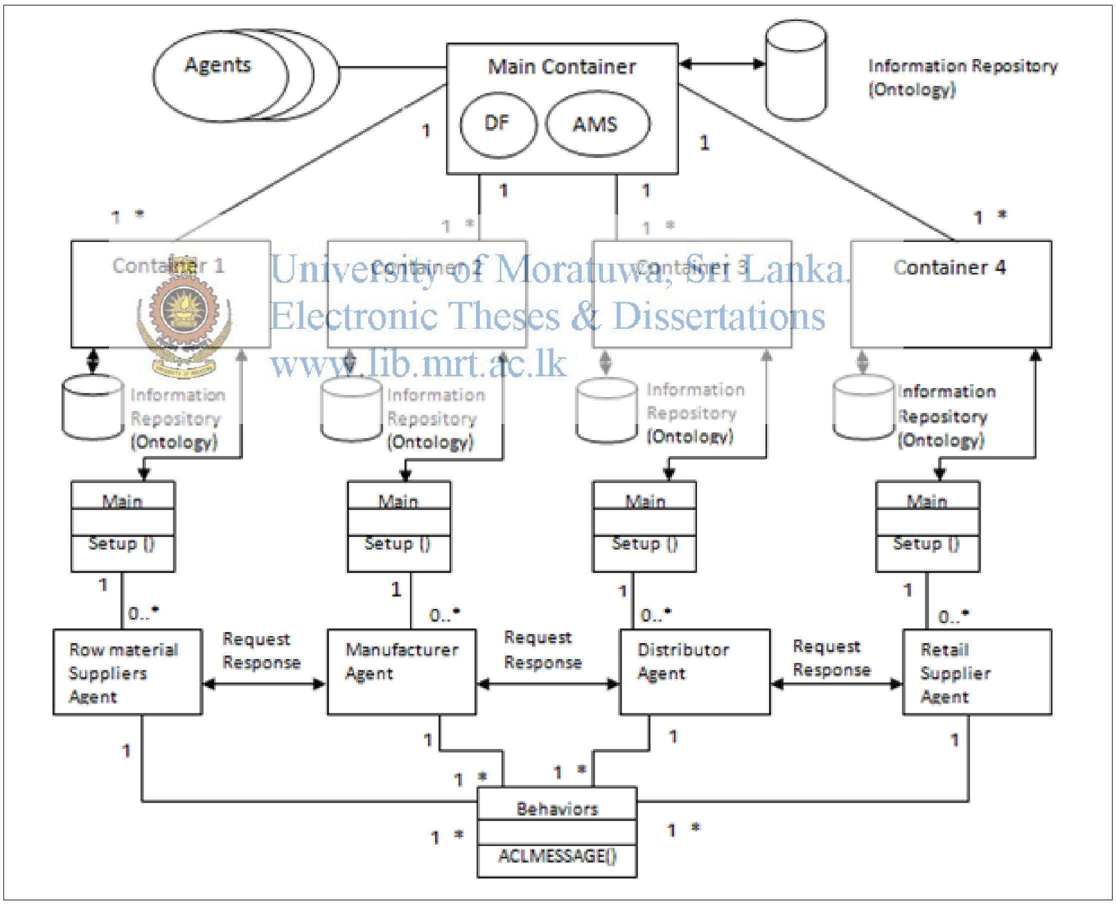


Figure 5.3: High-Level Architecture of SCM

According to the Figure 5.3 illustrates that different entities and their behaviors roles and responsibility. Each agent are attached to different containers and are authenticate by main container of the system. Java Agent Development Framework (JADE) has been used to implement containers and it is described in detail under implementation chapter. This diagram also explain that each agents are connected to sub containers to access their personal ontologies, and agent who are attached to main container also access domain ontology. Behaviors of the agents are highly coupled with connected ontologies, but main container has authority for agent administration such as transfer agent from one location to another (agent migration), suspend agent whenever it's necessary and terminate agent. Agent to agent communication happen via message passing that is included in the above diagram as request and response. However, a detail description about design diagrams such as class diagrams, sequence diagrams and state transition diagrams are included in Appendix B.

5.5 Agent Communication

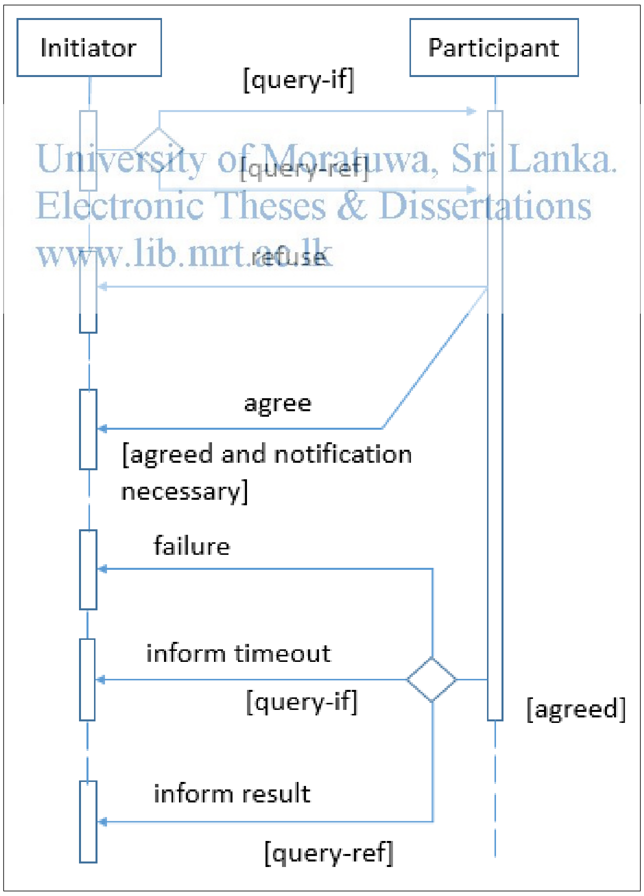


Figure 5.4: Sequence Diagram for Agent Communication

Agent communication is demanding tasks and sometimes it is complicated regardless of the application. Any complex system need correct design in relation to agent communication through message passing. In order to maintain proper negotiation mechanism the agents have to adhere with accepted protocol with other agents. FIPA stands for foundation for intelligence physical agent which is the protocol used for agent message passing. This allows an agent send a request to another agent (participant) for to perform certain action when specific condition is encountered. The target agent accept the request and has to wait considerable time to perform intended action. Finally, transmit a message to the initiator (requester) the requested action have been performed. This entire process is depicted in Figure 5.4 and detail implementation have demonstrated in implementation chapter.

5.6 Two Phase Development Architecture in SCM

The two component architecture have discussed early in this chapter to explain the importance of information sharing and collaborative planning. In spite of, two components architecture it essential to demonstrate the hierarchy of developments. There are two phases identified in SCM, including agent communication and administration phase and ontology management and report generation phase.

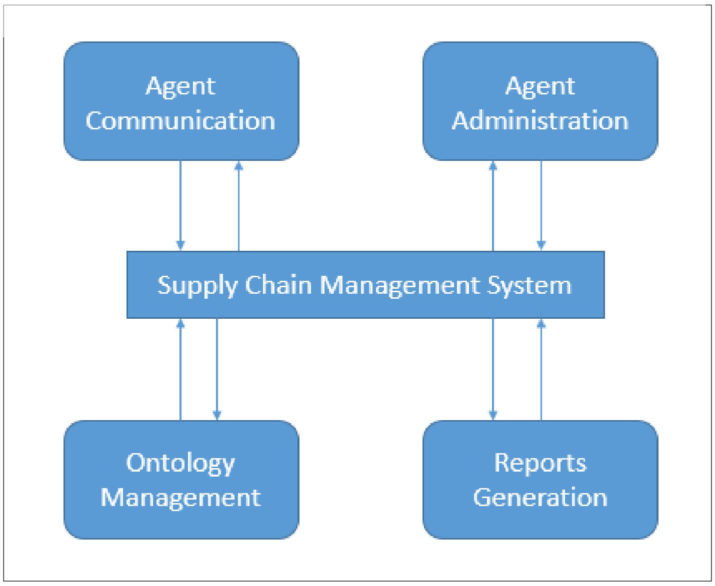


Figure 5.5: Development Architecture

The design of top level architecture in any application will compromise to successful implementation. As a deliverable of design phase the diagrams are drawn, and it includes description of the entire application architecture. The Figure 5.5 explains the entire system which include four components. The agent communication and administration have identified as parallel phase while ontology management and report generation phase have identified as two separate components. Agent communication and decision are highly coupled with ontology. However, results of agent communication have automated using report generation component. To improve the scalability of the solution each phases have been developed as separate components. This architecture is explained in details under implementation chapter.

5.7 Summary

This chapter described the overall picture of design architecture and the identification of different agents, modules (entities), roles and their responsibilities. Next chapter will prove how it has implemented using Java Agent Development Framework (JADE).

Chapter 6

Implementation of the MAS Solution for SCM

6.1 Introduction

Previous chapter illustrated the design architecture of supply chain complexity using multi agent technology. In this chapter our aim is to shift the emphasis from design to practice. Therefore, in this chapter will discuss the implementation of the complete software component which handle the complex communication in distributed environment. According to the design chapter the agents are shared and access the knowledgebase defined in personal ontologies and common domain ontology. However, there are physically two separate phases have been developed to align with design diagrams explained in the previous chapter. In general, one component is responsible for the implementation, administration and initialization of agents using JADE and java technology while other component is responsible for displaying web based dynamic reports of the communication and managing the ontology. Since a great deal of attention was paid to the implementation of ontologies using XML file system and MYSQL as knowledge base.

6.2 Agent Categorizations

To maintain simple implementation mechanism and to enhance modularization of components we have separated the agents in to six major categories as shown in Figure 5.2 in the chapter 5. Therefore, implementation have started with identified primary two components of raw material and manufacture agents. According to the supply chain domain manufacture have control over the most supply chain functions rather than raw materials suppliers.

In an open market we can find number of raw material suppliers producing of raw materials for manufacture venders. However, their raw materials prices have decided by manufacturers due to competition among the large number of suppliers. In order to start manufacturing depends on the requirement of customers. Customers have direct relationship with retail suppliers and they identify what is the customer wants and market trends. Finally, retailers shift the requirements to manufacturers and they demand the required quantities and varieties from raw materials suppliers. In any stage anomalies may arises due to poor communication among the supply chain therefore, agents have implemented according to the design architecture have mentioned in Figure 5.3 in chapter 5.

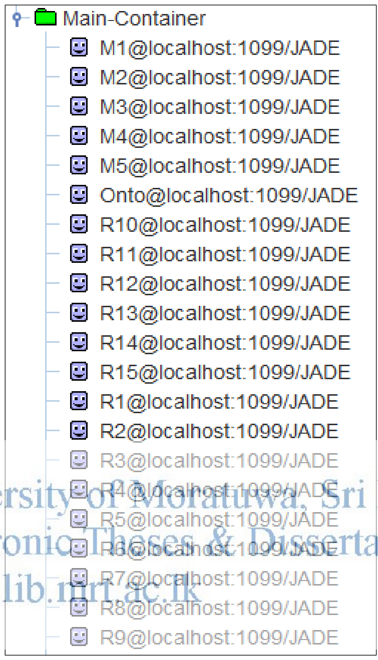


Figure 6.1: Manufacturer and Raw Materials Agents

The Figure 6.1 describes the number of agents are joined with communication process. A letter “M” is denoted manufacture agents while letter “R” denoted raw material agents. Further details about agent representation names/alias are described later in this chapter.

However, as an example manufacturer “M1” fails to manufacture products on time due to employees called as sick and problems with power failure. In such condition manufacture agents can communicate with both raw materials, transport agents, warehouse agents and retailers to overcome unnecessary delays in productions. The strong point of this multi-agent system is continuous communication and coordination to resolve anomalies. Therefore, all required parameters have been identified among the agents by the system such as number of employees have allocated for each batch, deadlines, daily input and output quantities and are reflected to other agents to prevent unwanted delays. The system have introduces the proper rules and ontologies for to shift the manufacturing duties from manufacture agent “M1” to “M2” according to their availability and capacity. The entire communication process is handle and monitored by message space agent and has the responsibility of handling the messages whenever necessary.

6.3 Alias of Agent Representation

When implementing agents using JADE framework is recommend to rename their original names to a short uniquely identifiable alias. Because, each agent category consists of substantial number of agents for to represent entire supply chain. No of agents used in this project is explained under Table 6.2 in this chapter. The process of agent message multicasting is identified as sender and receiver. It is proposed that to use agent id (AID) as a key components to uniquely identify and distinguish agents among other agents. As an example if there are fifteen raw material agent have implemented in the system their AID will range from R1 to R15.


```
Next available dates from manufacturer agent M2 is... 2015-04-02
Next available dates from manufacturer agent M2 is... 2015-04-07
Next available dates from manufacturer agent M2 is... 2015-04-06
The manufacturer agent M2 have agreed on 2015-04-07 to dispatch the products with T1
(INFORM
 :receiver  (set ( agent-identifier :name T1@localhost:1099/JADE  :addresses (sequence http://Manoj:
 :content  "New date schedule confirmation is received...!"
 :reply-with  T1@localhost:1099/JADE1432834300138 )
++++++++++++++++++++++++++++++++++++++++++++++++++ New date schedule negotiation is completed..+++++
```

Figure 6.2: Sender and Receiver

According to the Figure 6.2 there are two agents join with the communication process. The letter “M” represents manufacture agent while letter “T” denotes the transport agent. “M2” represent second manufacturer agent from manufacture agent category and “T1” is the first agent from transport agent category. In this stage “M2” sends the message to “T1” about scheduling transport dates. Therefore, “M2” is the sender while “T1” is the receiver. Additional details about agent alias are explained in Table 6.1.

Agent Name	Alias
Raw Material	R
Manufacturer	M
Transport	T
Human	H
Warehouse	W
Retailor	RE
Customer	C

Table 6.1: Agents Categories in Supply Chain with Alias

The benefits of the multi-agent systems by compared to humans is differ from their communication, negotiation and coordination rather fast and decisions are accurate. In order to maintain correctness of the communication their identification is must. The Table 6.1 defined agent category alone with their alias. These aliases are been used throughout this project for uniquely identification of agents.

6.3.1 No of Agents Join with Communication Process

Agent Category	Number of Agents
Raw Material	15
Manufacturer	5
Transport	3
Human	1
Warehouse	3
Retailor	15
Customer	30

Table 6.2: Number of Agents in Supply Chain

The Table 6.2 describes seven agent groups and number of agents are associated with under each category. According to the JADE framework is sufficient enough to create large number of agents under different containers of the system. Each container include one or more agents while main container has the facility of agent administration. Therefore, agent termination, migration, and suspension are the major activities associated with main container of the system.

In SCM different stakeholders are communicate in an operation, among others customer is the large category. Their inputs are strong enough to deploy butter-fly effect in any moment in the process because they consists of changing requirements all the time. Once the customer requirement have changed in one end will leads to larger modification in other end of the process in supply chain. However, this sudden change is accommodated with an introduction of multi-agent system in this implementation stage with advance feature provided by the framework.

6.4 Configuration and Initializations of Agents

Agent configuration and initialization have automated using JADE and java technology. One java class have been created for agent initialization. When system starts to run all agents have automatically configured and attached to their own containers according to the initialization parameters.


```
// Agents initialization

// R - Raw Meterials Suppliers
myAgent.getContainerController().createNewAgent("R1","pk1.RawMaterialAgent",null).start();
myAgent.getContainerController().createNewAgent("R2","pk1.RawMaterialAgent",null).start();
myAgent.getContainerController().createNewAgent("R3","pk1.RawMaterialAgent",null).start();
myAgent.getContainerController().createNewAgent("R4","pk1.RawMaterialAgent",null).start();
myAgent.getContainerController().createNewAgent("R5","pk1.RawMaterialAgent",null).start();
myAgent.getContainerController().createNewAgent("R6","pk1.RawMaterialAgent",null).start();
myAgent.getContainerController().createNewAgent("R7","pk1.RawMaterialAgent",null).start();
```

Figure 6.3: Agent Initializations

Figure 6.2 illustrates sample of raw material agent initialization. All initializations are grouped to a one java class with their alias names to maintain the explanation of agent administration. Agent initialization and administration are further discussed later in this chapter.

6.5 Agent Construction and Communication

One of the key component of multi-agent applications is agent communication. In fact, agents should be able to communicate with users, system resources and other agents to make decisions while performing coordination, negotiation and communication. In prior to agent communication their construction is demanding tasks. As mentioned in design chapter agents have to use specific mechanism to communicate. Throughout this research project agent creation and enable those to communicate while passing messages have taken important position rather than automating other functions of the system.

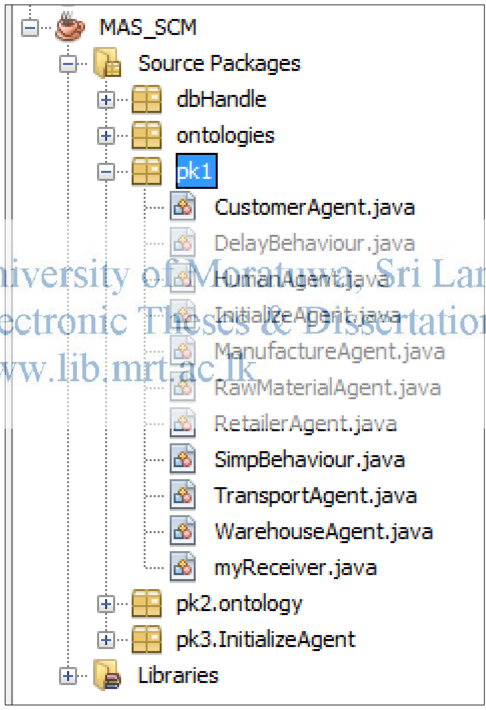


Figure 6.4: Agent Construction

As shown in Figure 6.4 this research project mainly developed using JAVA technology. NetBeans ID have used for agent construction and to operate their behaviors. Each agent have constructed using separate java class. A particular java class consist of diverse methods to manage different behaviors of an agents. In addition to agent construction, ontology management and agent administration also automated using JAVA technology. Agent creation, configuration, behaviors and communication have described in respectively Figure 6.5, 6.6 and 6.7.

6.5.1 Agent Creation

```
 * @author Manoj Lap
 */
public class TransportAgent extends Agent {

    MessageTemplate mt1 =
      MessageTemplate.and(
          MessageTemplate.MatchPerformative( ACLMessage.INFORM ),
          MessageTemplate.MatchSender( new AID( "H1",
                                      AID.ISLOCALNAME))) ;

    MessageTemplate mt2 =
      MessageTemplate.and(
          MessageTemplate.MatchPerformative( ACLMessage.INFORM ),
          MessageTemplate.MatchSender( new AID( "M1",
                                      AID.ISLOCALNAME))) ;
```

Figure 6.5: Create Individual Agents

With construction of empty java class and inherit/extend the properties from JADE library will turned java class to an agent. By default an agent have no any autonomous properties until they have incorporated with some kind of a behaviors. As shown in Figure 6.5 the transport agent have associated two templates namely “mt1” and “mt2”. The main purpose of the templates is to limit the response from other agents who are not requested to reply for certain request. Therefore, this template mechanism have controlled the message sending and receiving behaviors.

6.5.2 Agent Configuration

```java
@Override
protected void setup() { // Setup start
   super.setup(); //To change body of generated methods, choose Tools | Templates.
   addBehaviour(new TickerBehavior(this, 20000));
   addBehaviour(new delayBehaviorOnTrans (this, 30000));


    SequentialBehaviour seq = new SequentialBehaviour();
    addBehaviour(seq);

    ParallelBehaviour par = new ParallelBehaviour(ParallelBehaviour.WHEN_ALL);
    seq.addSubBehaviour(par);
```

Figure 6.6: Configure Individual Agents

The Figure 6.5 described how to create an agents and usage of message templates. As discussed in previous figure, an agent operation stat with the introduction of behaviors. The Figure 6.6 described different types of behaviors. JADE provide several ready-made classes to implement behaviors that are execute selected points in different times. These behaviors consist of abstracts methods to implement properties of behaviors. After construction of particular behavior, should be called under setup method. The setup method is default method provided by JADE framework and has the property to invoked methods/behaviors are created in java classes.

Every agent should have at least one behavior. So in this sense, one agent can have many behaviors activated concurrently. Behaviors are taken important position in prior to agent construction because, their operations are declared under different behavior code segments. Agent interaction start with execution of their behaviors. There are several types of behaviors supported by JADE framework including ticker behavior, one shot behavior, sequential behavior, parallel behavior and delay behavior. However, throughout this research project all of the behavior types have been incorporated to implement different interactions of an agents.

6.5.3 Message Sending Mechanism

```
public void handleElapsedTimeout(){
    try {
        // To w1 from t1
        Date w1date = null;
        ACLMessage toW1 = new ACLMessage(ACLMessage.INFORM);
        String sql = "SELECT DISTINCT(confirm_date),manu_aid FROM agents_delivery_schedule WH
        rs = da.SelectSql(sql);
        while(rs.next()){ // while if
        w1date = rs.getDate("confirm_date");
        String maid = rs.getString("manu_aid");
        if(w1date != null){
        toW1.setContent(w1date+"");
        toW1.addReceiver(new AID("W1", AID.ISLOCALNAME));
        if(toW1 != null){ // Check the content is null or no
        send(toW1);
        System.out.println("The sender is "+myAgent.getLocalName()+" and the confirmed dates 
        block(2000);
        } // end null
```

Figure 6.7: Communicate Individual Agents

A fundamental characteristics of multi-agent system is agent communication and interaction. This has achieved through message passing and understanding the each other. In other words, continuous communication paradigm is based on asynchronous message passing. The crucial aspect of the communication is, agents agree on the format of the message and the semantics of the content. JADE uses FIPA standards in communication. In the absences FIPA standards multi-agent system will not be popular application in industry automation of business functions. The agent message consist of three parts including sender, receiver and the content. The output of the message and interaction of the receiver have been discussed in evaluation chapter.

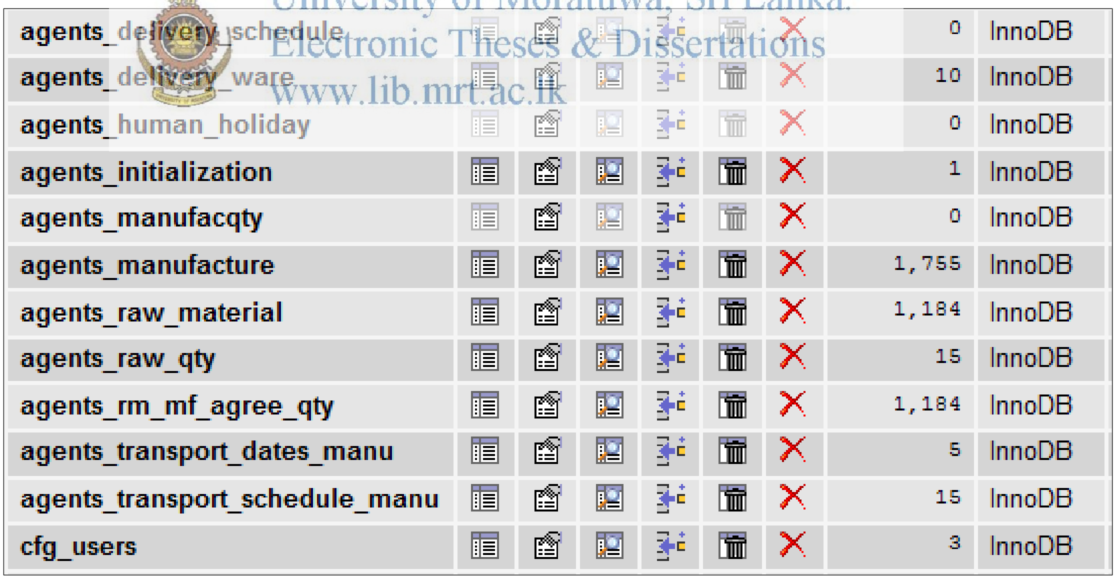
In addition to the parts of the message another mechanism have to apply before sending it to the receivers. The communicative act is the primary mechanism, and it have been used to implement what the sender intends to achieve by sending the message. The communicative act also called as PERFORMATIVE [44] in JADE framework. To exchange the message agent have to use certain communication language. The first agent communication language is KQML and it have been discussed in chapter three. However, currently the most suited and widely used language is FIPA.

6.6 Common Domain Ontology and Personal Ontology

Every agent have to access the knowledge base and their actions are highly coupled on the defined ontology. Throughout this project there are two types of knowledge bases have declared respectively personal ontology and common domain ontology.

6.6.1 Personal Ontology

Globally collective decisions are based on common domain ontology which has constructed using XML format while personal ontology have stored in MYSQL database. In connection with Figure 5.3 in design chapter has clearly indicates that one ontology is accessed by the agents through main container while other ontologies are accessed by an agents through sub containers are connected to main container. The Figure 6.11 in chapter 6.7 described detail description about main container and its operation.



agents_delivery_schedule	0	InnoDB
agents_delivery_ware	10	InnoDB
agents_human_holiday	0	InnoDB
agents_initialization	1	InnoDB
agents_manufacqty	0	InnoDB
agents_manufacture	1,755	InnoDB
agents_raw_material	1,184	InnoDB
agents_raw_qty	15	InnoDB
agents_rm_mf_agree_qty	1,184	InnoDB
agents_transport_dates_manu	5	InnoDB
agents_transport_schedule_manu	15	InnoDB
cfg_users	3	InnoDB

Figure 6.8: Personal Ontology (MYSQL Database)

Agent communications are fast and accurate than humans. In order to provide rapid response to any queries from the supply chain the agents have to access their ontologies without any disturbance. Therefore Figure 6.8 indicates that snap shot of MYSQL database with advanced features such as indexing and more than 150 multiple concurrent connections in any given moment. Therefore MYSQL database is sufficient to provide concurrent access to any number of agents throughout the network with zero failures.

6.6.2 Common Domain Ontology

The common domain ontology is another aspect of the knowledge base. This has constructed using XML version 1.0 as novel technique to make information readable of using any technology. Therefore, regardless of the programming language any agent in the network will able to read and write the contents to XML files.

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<agentConfiguration>
    <itemName>Books</itemName>
    <itemPrice>10.00</itemPrice>
    <itemQuantity>650</itemQuantity>
</agentConfiguration>
```


Figure 6.9: XML Content

There are some information should be accessible to others in supply chain. As an example description of the product, price and available quantities are should be visible to other agents in supply chain. The Figure 6.9 described part of the XML content is available to other agents for communication. For read and write the XML content need special technique. Therefore, it has used JAXB, it is an open source library which provide the mechanism of converting XML content to JAVA objects and it is further explained in Figure 6.10.

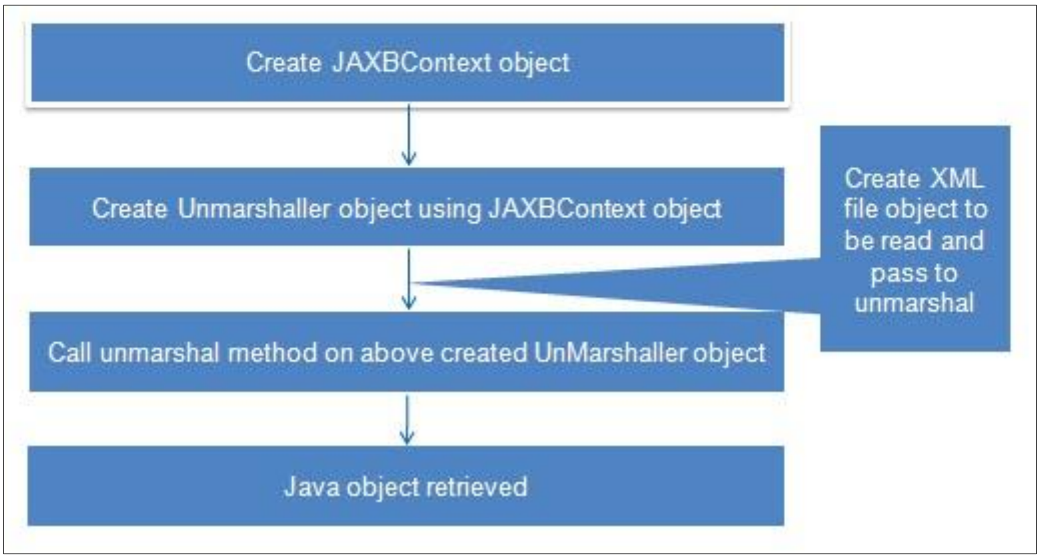


Figure 6.10: JAXB Architecture

JAXB stands for JAVA architecture for XML bindings. It include properties for to create, read and write the XML content. XML and JAVA technology are recognized as perfect building blocks for develop web services through internet. The Figure 6.10 described the hierarchy of creating JAVA XML object using methods of Marshalling and Unmarshalling. Marshalling convert the java object into XML representation while unmarshalling is the process of creating XML content back to JAVA object.

6.7 The Main Container and its Features

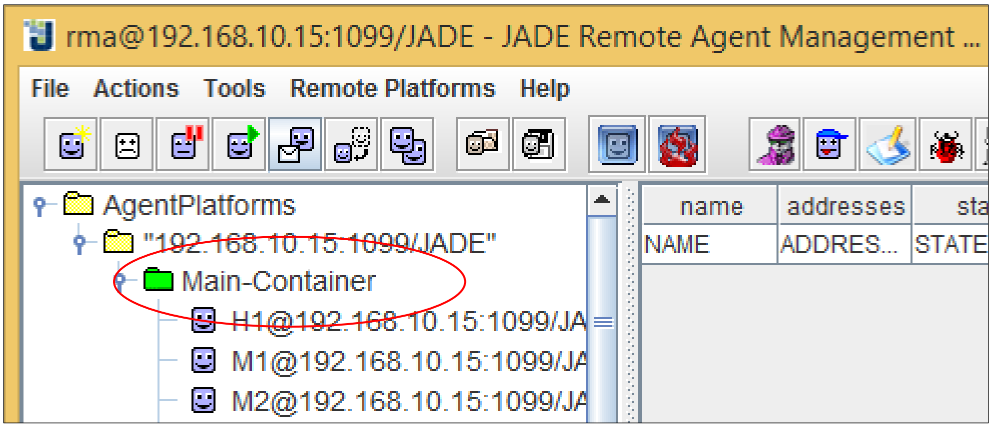


Figure 6.11: Main Container Initializations

The importance and the relevance of main container in JADE framework is exceptional because all other containers are connected and authenticated by main container. The Figure 6.11 shows that when agent application is active then the main container will automatically started. This container is assigned default IP address of the computer and all other containers are connected via the default IP address have received by the main container. Therefore, any sub container have implemented in network will able to connect with main container alone with their local IP addresses. In this connection, regardless of the application type, if any system uses JADE framework the main container will be the backbone of the entire system.

6.8 JADE Framework and Agent Communication

JADE is a software framework written in JAVA language. It streamlines the multi-agent systems through middleware component according to foundation for intelligent physical agent (FIPA) specification [51]. JADE based applications are platform independent and can be execute in any operating system. JADE framework is consists of remote graphical user interface (GUI) and it helps to administrate the agent at a runtime without affecting their initial status. Agent administration will be discussed later in this chapter. JADE provide simple and powerful composition model for peer to peer agent communication. In addition, to align with Figure 5.4 in design chapter need specific mechanism to demonstrate the agent communication. Agent communication happens via message passing techniques and it has depicted in Figure 6.12.

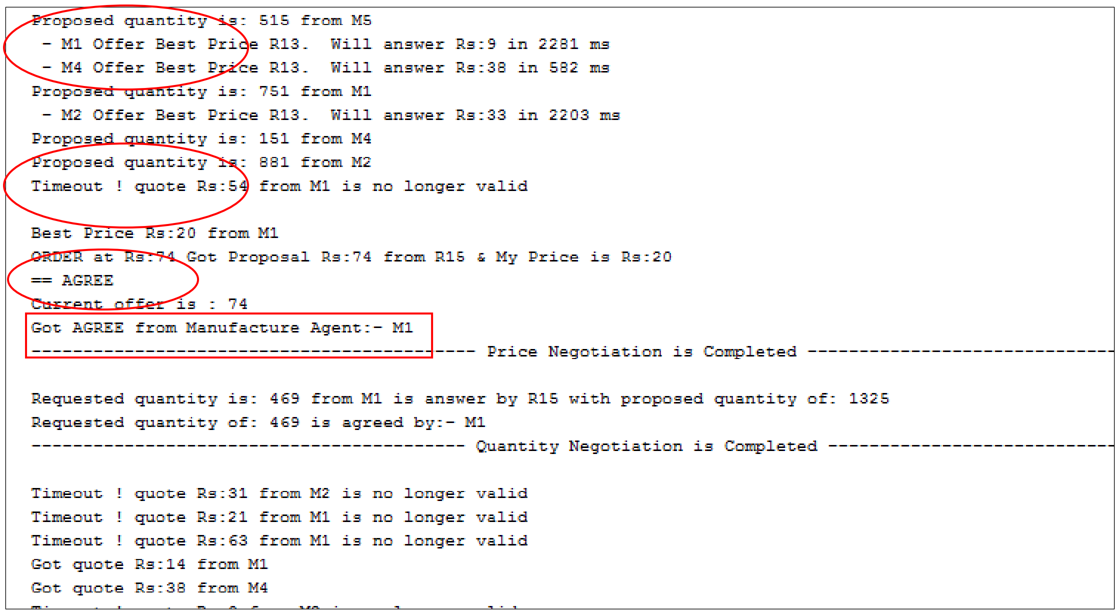


Figure 6.12: Agent Communication

According to the sequence diagram (Figure 5.4) in design chapter the initiator is represented in Figure 6.12 as manufacturer agents (M1, M4). These manufacturer agent send multicast request to raw material agents (participant/responder) to perform an action. The intended raw material agents (R15) is wait certain moment and start to respond when the anticipated condition is achieved. Figure 6.8 is shows some of the request message are not committed due to timeouts (timeout from M1). When the received message met the proposed condition both parties will come to an agreement. After completion of the agreement the recipient have to feedback to the initiator (manufacturer M1). This communication will meet an end with the received message from recipient. Finally, the square is highlighted that the received confirmation message from the responder by the initiator.

6.9 Agent Administration

Agent administration is a prominent, because sometimes when completion of the given tasks for an agent need to termination. So in this sense, the amount of memory is allocated for an agent is will be a waste, if the intended agent is active even after completion of the tasks. There are some instance agent need to migrate from one platform to another platform. As an example a retailer agent is operate in one location and shifted to some other country due to cheap labor and some financial constraints. In such a situation agent platform have to change. This have achieved using GUI component provided by JADE runtime.

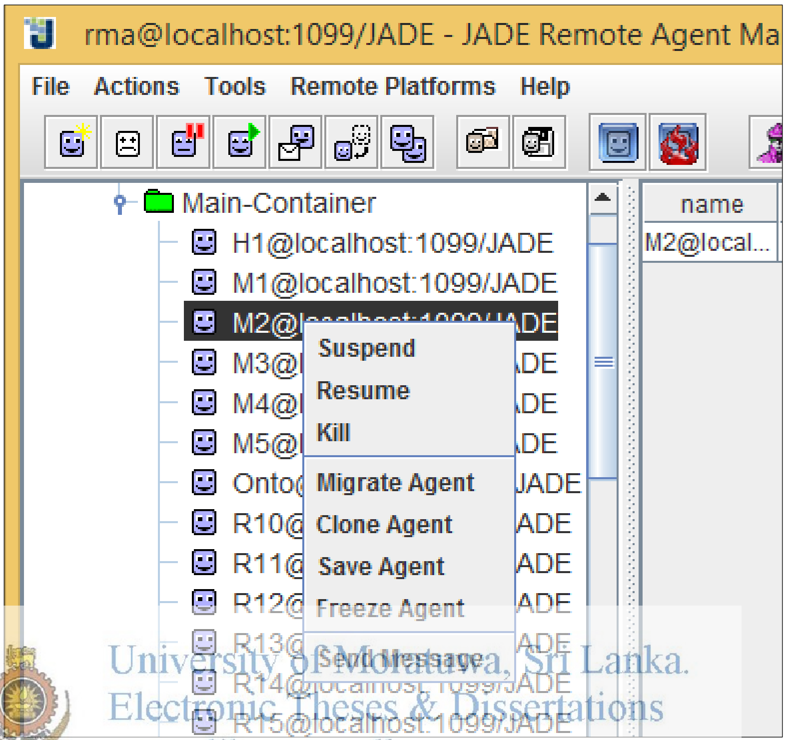


Figure 6.13: Agent Administration Tool

Main container has the agent administration facility while other sub containers are registered with the main container have limited agent administration facility. Sub containers can manage only agents are registered with them. The Figure 6.13 depicted that available features of JADE runtime in relation to agent administration.

6.10 Ontology Management

To align with design diagram (Figure 5.5) shown in design chapter ontology management is important regardless of the application domain. In this application ontology management have identified as critical tasks. Agents are concurrently read and update ontology to generate corporative decisions.

```
ACLMessage msgR = receive(receiveFormation);

if(null != msgR){
        messageReceived  = true;

        //The agent M3 reply Message from Warehouse Agent...! to T3 (need to correct)....
        System.out.println("The agent "+agents[i].getName().getLocalName() +" reply " + msgR.getContent()+
        getAids = agents[i].getName().getLocalName();

        String query = "SELECT tr_aid,tr_sdate,tr_edate,tr_time,tr_qty FROM agents_transport_schedule_manu
        rs = da.SelectSql(query);
    try {
        while(rs.next()){

            trAid = rs.getString("tr_aid");
            sdate = rs.getDate("tr_sdate");
            edate = rs.getDate("tr_edate");
            stime = rs.getTime("tr_time");
            trQty = rs.getShort("tr_qty");

            String updateQuery = "UPDATE agents_transport_schedule_manu SET read_status='"+readStatusUpdate

    try {
        da.UpdateSql(updateQuery);
    } catch (SQLException ex) {
        ex.printStackTrace();
```

Figure 6.14: Ontology Read and Update

As discussed early in this chapter there are two types of ontologies have used in this application. Ontology is worked as backbone to any system while their usage is different depend on the application domain. The Figure 6.14 shows that the ways of accessing and updating ontology constructed in MYSQL database. The first circle explained an agents are accessing the information available in knowledge base. The second circle highlighted that an agents are updating the ontology with collaborative decisions.

In addition, there is another mechanism have implemented to manage ontology. Supply chin functions are distributed in different networks. However, modifications to ontology happens in distributed environments. Therefore, specific web component have constructed in parallel to agent communication. This web component have facility to update agent knowledge base in real time with access constraints. The Figure 6.15 explained the web component GUI is available for ontology alteration.

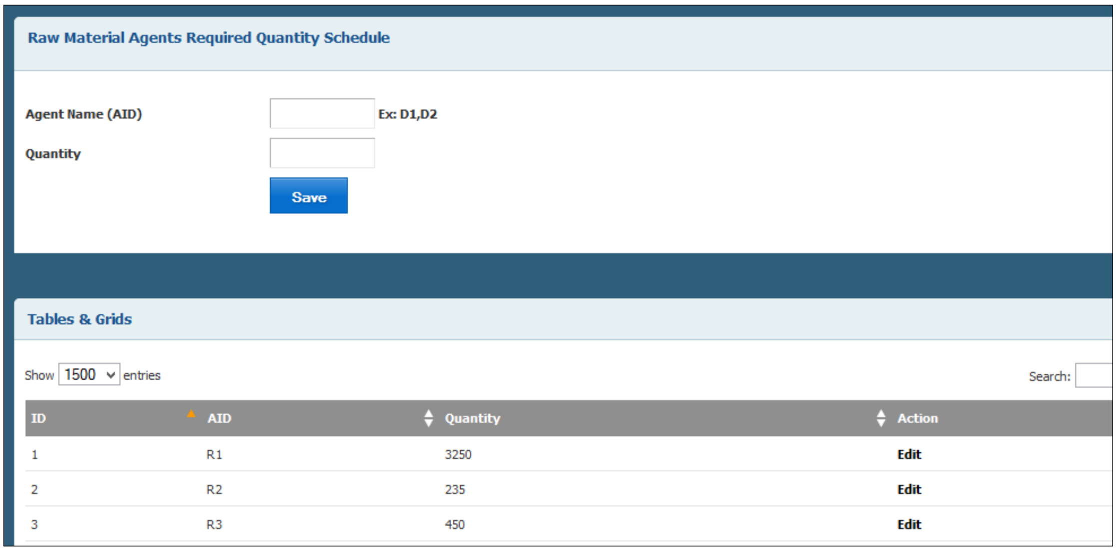


Figure 6.15: Ontology Management GUI

As said in design chapter, ontology management is one of the core module in this application. The Figure 6.15 illustrated that GUI tool have constructed for ontology management through web application.

6.11 Dynamic Reporting Mechanism for Real Time Communication

Supply chain management is complex and distributed through the network. SCM decision always need to be fast and accurate. As mentioned early in this chapter, agents are designated to derive accurate and fast decision according to their defined knowledge base/ontology. In order to continuous growth of the business the management have to derive correct decision of enterprise resources. This dynamic report mechanism provide real time reports about current situation and fast records through online. This has automated using web technology including PHP, JQuery, and Java script. As mentioned in Figure 5.5 in design chapter this is the last component have constructed to complete this research project.

The data is derived from knowledge base (XML, MYSQL database) which continuously updated by an agents through the network. The report mechanism have been introduced a graphical tool to represent information in different viewpoints. This has mentioned and described alone with screen shots in evaluation chapter.

6.12 Hardware and Software Requirements

In any software project it is a good practice to mention the required software and hardware is needed to run the application. In this research projects the completed software is constructed using different software technologies. Under this topic is basically intended to discuss the required hardware and software needed to run the application. In some extent the required software for development environment have discussed at the beginning of this chapter. The Table 6.3 discussed in detail about software, hardware and operating system requirements.

Software Requirement for Development Environment	
	JAVA IDE, JADE, NetBeans IDE, PHP, JQuery, Java Script, CSS and other HTML 5 components
Software Requirement for System Deployment	
	JAVA runtime (1.8)
	Apache Server 2.2 or higher
	Hyper Text Pre Processor (PHP) 5.3 or higher
	MYSQL Server 5.1 or higher
Hardware Requirement for System Deployment	
	Core I, 5 or higher processor
	500 GB Hard drive
	8GB RAM
Operating System	
	The system have developed using free and open sources software therefore, system can execute both Windows and Linux platforms.

Table 6.3: Hardware and Software Requirement

6.13 Summary

The heart of this research project is to provide working software components with intended outputs. In this chapter we have explained different implementation strategies aligned with design diagrams shown in design chapter. One phase of this application is agent communication and administration while other component includes ontology management and dynamic reports.

Detail discussion was taken place about-multi agent communication and ontology management. The next chapter will discussed supply chain management as a real world problem and importance of novel approach. In addition, it also included a comparison of features about traditional systems against multi-agent systems.

Chapter 7

A Real World Problem Using Novel Approach

7.1 Introduction

According to the previous chapter in implementation stage clearly shows that the steps are used to automate the supply chain functions using multi agent technology. However, throughout this chapter is intended to describe the real world problem and new technology alignment to provide smart solution in supply chain management. The ultimate goal of every organization is to earn maximum profit in business regardless of the type of the operation. In supply chain control the flow of management resources for to maintain the business operation and to earn high profitability.

Overall performance is depend on the complex and dynamic interaction with the components. The modern concept of supply chain management has wide definition that incorporates business logistics and physical distribution. In addition to that the required resources are not isolated they are distributed among countries. However, to make final product all of such a resources need combination in right time and right quantities.

It is a question of larger spectators why supply chain is a real world problem among other business operation. The basic idea of the supply chain is to sell goods to consumers while making maximum profitability in competition with other business organizations. One of the most interesting aspect in supply chain is dynamic interaction among stakeholders. This is due to the circumstance that supply chains are multi echelon systems which deliver goods to customers from the point of origin, throughout deferent departments and organizations. However, there is a tendency any stakeholder can make mistakes while operating their business strategies. Therefore, one mistake in one end of the supply chain will propagate a huge results (butter-fly effect) in entire business operation.

7.2 Use of Novel Approach

Since a supply chain is dynamic and chaotic and its structure must be flexible to cope with changing demand in market. A supply chain must change the pattern without any effect to gain competitive advantage in business operation. Without automation of the functions of a supply chain if we introduced humans instead of intelligent agents the results must not be make high profits and sometimes it could be unpredictable. The humans are not perfect in communication to some extent and have limitation when adopting sudden changes in the environment. However, if we used traditional object oriented architecture has the limitation in the area of information distortion, delay and not compete with bullwhip [18] effect.

Therefore, the needfulness of advanced framework with accepted communication standards is highly beneficial for a successful supply chain management.

7.3 A Difference between Traditional Systems and Novel Approach

Our novel approach consists of multi-agent technology which is one of the discipline in artificial intelligence. The Table 7.1 have described the comparison of characteristics in traditional systems and multi-agent systems. The basic idea is graphically depicted in Figure 7.1 which consists of multiple components and their interactions.

Traditional Systems	Multi Agent Systems
Hierarchy of Components	Network of Agents
Sequential Processing	Parallel Processing
Top-Down Instructions	Negotiations and Trade-Offs
Centralized	Distributed
Data Driven	Knowledge Driven
Predictable	Self-Organized
Reduced Complexity	Increase Complexity
Result Driven Decisions	Collective Decisions
Consume Resources	Limited Resources
Local Results	Outstanding Global Results

Table 7.1: Traditional Systems and MAS

According to the given properties in Table 7.1 have indicated that the limitations associated with traditional systems. In this novel approach is increased the complexity in communication. However, that leads to find new pathways, smart solutions, reduce delays and increased emergent properties while providing maximum satisfaction to its stakeholders.

7.4 Roles of the Agents in Novel Approach

Novel approach is started with creating an intelligent agents to represents stages in supply chain management while enabling communication, coordination and negotiation among them. Each agent have assign to complete small tasks and enable to access ontologies defined in personal ontology and common domain ontology. When agents have started to communicate with other agents their decisions are updated in ontology files. Any given time there are more than ten agents active in the communication process and, with the completion of the tasks they can leave the message space at any time. More definition about simultaneous active agents are defined in Figure 7.1 with compared to traditional systems. Agent communications, roles, decisions, tasks and responsibilities are explained in details under evaluation of the new solution chapter.

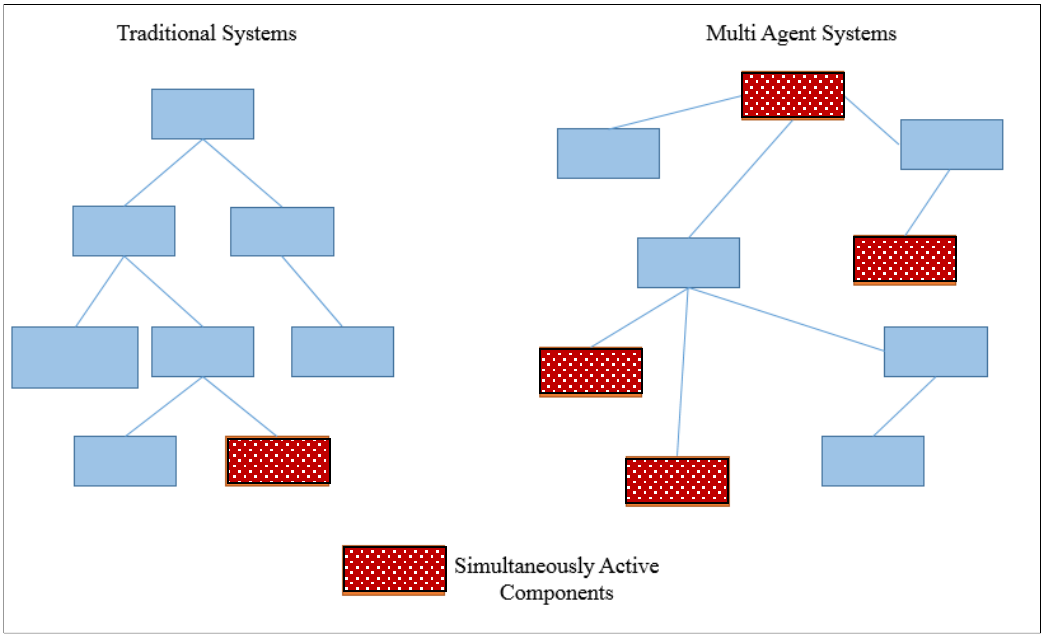


Figure 7.1: Simultaneously Active Components

The Figure 7.1 indicates that simultaneously active components in traditional systems and multi agent systems. The traditional systems maintain the hierarchy of communication while multi-agent system is used network of components. If any single point failure is occurs in traditional systems will face to entire communication system to a not responding state. However, with more redundant communication links indicate that continuous communication in multi-agent systems by compared to traditional systems. Therefore, this novel approach is committed to the continuous communication mechanism for the successfulness in supply chain.

7.5 Summary

In general, this chapter has illustrated that the definition of real world problem and why SCM is denoted as real world problem. Instead, it also explained the different between traditional system and the importance of novel approach. The next chapter is intended to explain step by step approach of the new system evaluation. In addition, it also planned to provide detail evaluation in each steps of the entire software component. In the evaluation chapter will discuss multi-agent features and resulted outputs to satisfy different stakeholders of the system.

Chapter 8

Evaluation of the New Solution

8.1 Introduction

In previous chapter explained that detail description about real world problem, and also discussed the importance of novel approach in communication. However, in this chapter is planned to provide real world emphasis on supply chain as new and smart solution in order to satisfy stakeholders of the system. Up to this moment entire research project have focused the subjectivity of the application. Throughout this chapter it changed to prove the objectivity of the application.

In any research project the evaluation stage is critical because it will mapped the resulted outcome with stakeholder satisfaction. Therefore, this chapter have taken important strategies to verify outcomes with customer requirements. In addition to that, this chapter is evident how multi-agent features are affected to supply chain functions to obtain diverse solutions. In particular, early negotiation of agents, agent interaction, interference of human agent, change in business process, and real world data have discussed as main evaluation points.

8.2 Beginning of the Negotiation in Supply Chain Management

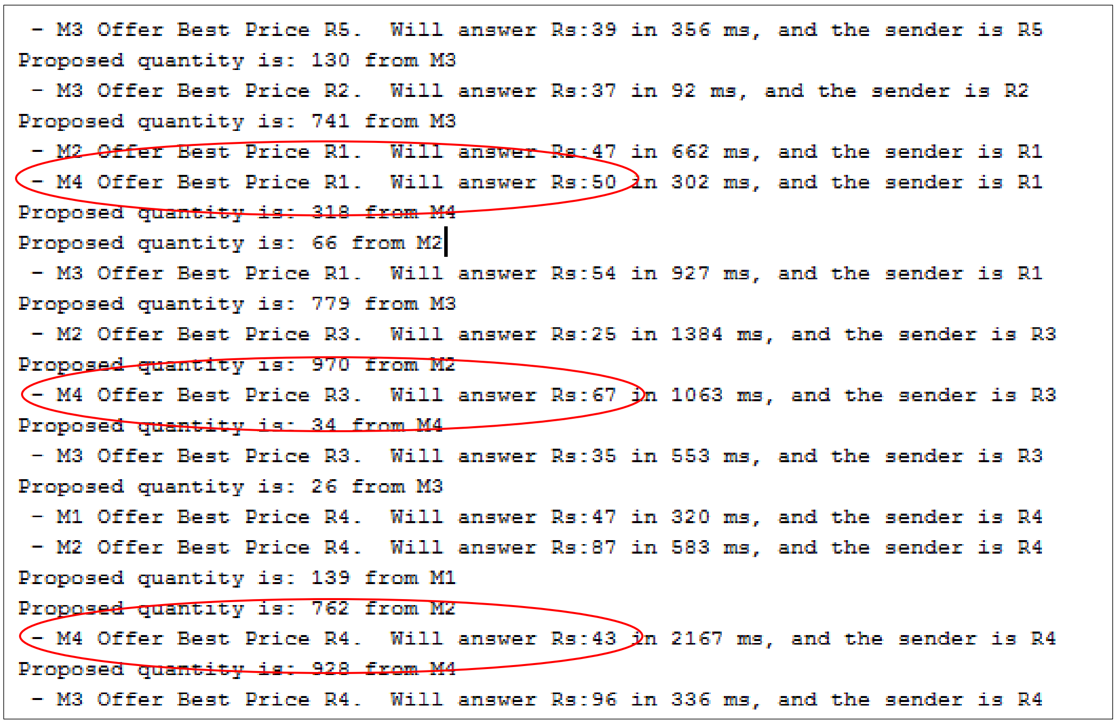
As discussed in previous chapters, this research project is mainly consist of multi-agent technology, features, and how it benefit for the smooth functioning of continuous business process. Therefore, negotiation, communication and coordination are some of the key features in any multi-agent application is intended to demonstrate. The Figure 8.1 evident that, how agents are start to negotiate with iteration of others.

```
Raw Material Supplier Agent R2 Got Demand from Manufacturer Agent.
--------------------------------------------------------------------------------
Raw Material Supplier Agent R10 Got Demand from Manufacturer Agent.
--------------------------------------------------------------------------------
Raw Material Supplier Agent R5 Got Demand from Manufacturer Agent.
--------------------------------------------------------------------------------
Raw Material Supplier Agent R12 Got Demand from Manufacturer Agent.
--------------------------------------------------------------------------------
Raw Material Supplier Agent R4 Got Demand from Manufacturer Agent.
--------------------------------------------------------------------------------
Raw Material Supplier Agent R1 Got Demand from Manufacturer Agent.
--------------------------------------------------------------------------------
Raw Material Supplier Agent R9 Got Demand from Manufacturer Agent.
--------------------------------------------------------------------------------
Raw Material Supplier Agent R3 Got Demand from Manufacturer Agent.
--------------------------------------------------------------------------------
Raw Material Supplier Agent R15 Got Demand from Manufacturer Agent.
--------------------------------------------------------------------------------
Raw Material Supplier Agent R7 Got Demand from Manufacturer Agent.
--------------------------------------------------------------------------------
Raw Material Supplier Agent R8 Got Demand from Manufacturer Agent.
--------------------------------------------------------------------------------
Raw Material Supplier Agent R13 Got Demand from Manufacturer Agent.
--------------------------------------------------------------------------------
Raw Material Supplier Agent R11 Got Demand from Manufacturer Agent.
--------------------------------------------------------------------------------
```

Figure 8.1: Start of the Negotiation

Agent iteration is one of the key feature in multi-agent application. In this research project agent communication start with receiving raw material demands form manufacturer agents. There are fifteen raw material agent (R1-R15) have declared, but in this Figure 8.1 shows only thirteen raw material agents. When communication start there are some active agents. However, there is no guaranteed that always every agent will join with the communication because their behaviors are dynamic. In contrast, agents are active and they start communication with certain delay.

The main purpose of this dynamic behavior of early negotiation is to complete the tasks within limited resources. The figure 8.2 have depicted that only five raw material agents (R1-R5) in active state. However, the Figure 8.3 described fifteen raw material agents (R1-R15) are communicated with manufacture agents. There are only five manufacturer agents have been initialized on both instances. This comparison is intended to provide when competition is increased the benefit will receive by manufacturers. Therefore, initial cost of the product have reduced considerably.



```
 - M3 Offer Best Price R5.  Will answer Rs:39 in 356 ms, and the sender is R5
Proposed quantity is: 130 from M3
 - M3 Offer Best Price R2.  Will answer Rs:37 in 92 ms, and the sender is R2
Proposed quantity is: 741 from M3
 - M2 Offer Best Price R1.  Will answer Rs:47 in 662 ms, and the sender is R1
 - M4 Offer Best Price R1.  Will answer Rs:50 in 302 ms, and the sender is R1
Proposed quantity is: 318 from M4
Proposed quantity is: 66 from M2
 - M3 Offer Best Price R1.  Will answer Rs:54 in 927 ms, and the sender is R1
Proposed quantity is: 779 from M3
 - M2 Offer Best Price R3.  Will answer Rs:25 in 1384 ms, and the sender is R3
Proposed quantity is: 970 from M2
 - M4 Offer Best Price R3.  Will answer Rs:67 in 1063 ms, and the sender is R3
Proposed quantity is: 34 from M4
 - M3 Offer Best Price R3.  Will answer Rs:35 in 553 ms, and the sender is R3
Proposed quantity is: 26 from M3
 - M1 Offer Best Price R4.  Will answer Rs:47 in 320 ms, and the sender is R4
 - M2 Offer Best Price R4.  Will answer Rs:87 in 583 ms, and the sender is R4
Proposed quantity is: 139 from M1
Proposed quantity is: 762 from M2
 - M4 Offer Best Price R4.  Will answer Rs:43 in 2167 ms, and the sender is R4
Proposed quantity is: 928 from M4
 - M3 Offer Best Price R4.  Will answer Rs:96 in 336 ms, and the sender is R4
```

Figure 8.2: Raw Material Agent Communication in Limited Competitive Environment

As discussed in implementation chapter the letter "M" is denoted as manufacturer agent. There are five manufacturer agents active in this stage. In each instance of communication, manufacture agents have authority to provide offers for raw material agents to provide competitive prices. In this instance "M4" have received three different prices from respectively "R1", "R3" and "R4". Among received prices "R4" is the lowest and it is Rs: 43. The key idea in this communication is depicted and described in Figure 8.3.

```
 - M4 Offer Best Price R2.  Will answer Rs:17 in 1431 ms, and the sender is R2
Proposed quantity is: 978 from M5
Proposed quantity is: 45 from M1
Proposed quantity is: 254 from M4
 - M2 Offer Best Price R14.  Will answer Rs:76 in 2586 ms, and the sender is R14
Proposed quantity is: 408 from M2
 - M4 Offer Best Price R14.  Will answer Rs:33 in 207 ms, and the sender is R14
Proposed quantity is: 768 from M4
 - M1 Offer Best Price R14.  Will answer Rs:20 in 525 ms, and the sender is R14
Proposed quantity is: 290 from M1
 - M5 Offer Best Price R14.  Will answer Rs:73 in 1990 ms, and the sender is R14
Proposed quantity is: 156 from M5
 - M2 Offer Best Price R15.  Will answer Rs:41 in 820 ms, and the sender is R15
Proposed quantity is: 748 from M2
 - M1 Offer Best Price R15.  Will answer Rs:16 in 611 ms, and the sender is R15
Proposed quantity is: 858 from M1
 - M5 Offer Best Price R15.  Will answer Rs:6 in 74 ms, and the sender is R15
 - M4 Offer Best Price R15.  Will answer Rs:68 in 682 ms, and the sender is R15
Proposed quantity is: 949 from M4
Proposed quantity is: 806 from M5
 - M2 Offer Best Price R4.  Will answer Rs:1 in 1644 ms, and the sender is R4
```

Figure 8.3: Raw Material Agent Communication in Competitive Environment

As mentioned in Figure 8.2 the received lowest price is Rs: 43. However, the Figure 8.3 illustrated that the manufacturer agent have received Rs: 17 as the lowest price from “R14” in second instance of communication. Therefore, when environment is competitive and dynamic the result of the communication is competitive. The key idea of an interaction is to obtain lowest price from raw material agent by taking advantage of their competition.

When number of raw material agents are high in the communication process the initial price for their products will automatically reduce. In such situation manufacturer agent will receive competitive price and control the business function with higher authority. In supply chain the manufacturers have control the production stage while transport agents have control from production stage to distribution stage. Finally, retailers have the authority of controlling selling stage. However, different evaluation strategies are included throughout this chapter to provide evidence of outcomes. The results of continuous communication have clearly summarized in Table 8.1.

First Instance in Communication with Five Raw Material Agent in Active			Second Instance of Communication with Fifteen Raw Material Agents in Active		
Manufacture Agent AID	Raw material Agent AID	Price Rs:	Manufacture Agent AID	Raw material Agent AID	Price Rs:
M4	R1	50	M4	R2	17
M4	R3	67	M4	R3	67
M4	R4	43	M4	R15	68

Table 8.1: Competitive Price Schedule of Manufacturer Agent "M4"

As discussed under Figure 8.2 and 8.3, the received results have depicted in Table 8.1 for clear understanding. It is obvious that "M4" have received the lowest price among other raw material suppliers in more competitive environment. The same scenario have described using graphical notations at the end of this chapter.

8.3 Continuous Negotiation for Commitment

An agent is start to communicate with some other agent will finished the negotiation process with completion of tasks. In multi-agent applications every agent will not commit for the completion of tasks, because the particular tasks will complete by some other agent (dynamic/emergent). Therefore, completion of the tasks have taken important step in negotiation. As said early in this chapter, manufacturer agents are received prices for the products.

However, received prices will not commit for the completion of the tasks, there are some other parameters have to complete. In such instance manufacture agent have to receive intended quantities to for the completion of the negotiation process. If they not received the required quantities, the communication process will start to continue until bath parties come to a conclusion about product quantities. In some occurrence, the lowest price offered raw material agent will not be able to supply required quantities. This scenario have sown in Figure 8.4.

```
Best Price Rs:6 from M2
ORDER at Rs:75 Got Proposal Rs:75 from R3 & My Price is Rs:6
Got quote Rs:50 from M1
Got quote Rs:50 from M4

Best Price Rs:51 from M4
== AGREE
Current offer is : 75
Got AGREE from Manufacture Agent:- M2
------------------------------------------- Price Negotiation is Completed -----------

Timeout ! quote Rs:94 from M4 is no longer valid
Timeout ! quote Rs:46 from M1 is no longer valid

Best Price Rs:38 from M1
ORDER at Rs:8 Got Proposal Rs:8 from R8 & My Price is Rs:38
Requested quantity is: 1465 from M2 is answer by R3 with proposed quantity of: 450
Quantity is not sufficient...R3
```

Figure 8.4: Agreement and Refuse from Manufacture Agents

In supply chain business activities are self-motivated. In one occasion venders are agreed with purchases while other occasion they completely refuse the business activity due to some reason. Therefore, satisfying all stakeholders in the business process is curtail. However, the solutions have provided by the conventional systems are found difficulties to address changing requirements of the business venders. Therefore, continuous communication as a smart feature in multi-agent technology have addressed this limitation in any circumstances. The Figure 8.4 discussed the sudden change in the communication. At the early stage in discussion the manufacture agent “M2” have received Rs: 75 from raw material vender “R3”.

As a result of price negotiation they came to an agreement and it is circled (second circle) in the Figure 8.4. To complete the task bath agents have to confirm available and required product quantities. In this situation the raw material vender “R3” does not have required quantity is proposed by “M2”. Therefore, communication is committed between “M2 and R3” with no results. The benefit of the continuous communication and results obtained have depicted in Figure 8.5.

```
Got Proposal Rs:53 from R1 & My Price is Rs:19
Got quote Rs:99 from M3
== REFUSE
Current offer is : 8
Got REFUSE from Manufacture Agent:- M1

Best Price Rs:58 from M3
Raw Material Supplier Agent R8 Got Demand from Manufacturer Agent.
---------------------------------------------------------------------------------
 - M3 Offer Best Price R8.  Will answer Rs:49 in 1158 ms, and the sender is R8
Proposed quantity is: 837 from M3
 - M4 Offer Best Price R8.  Will answer Rs:54 in 943 ms, and the sender is R8
Proposed quantity is: 606 from M4
 - M5 Offer Best Price R8.  Will answer Rs:17 in 2160 ms, and the sender is R8
Proposed quantity is: 415 from M5
Requested quantity is: 455 from M1 is answer by R8 with proposed quantity of: 200
Quantity is not sufficient...R8
== AGREE
Current offer is : 53
Got AGREE from Manufacture Agent:- M2
----------------------------------------- Price Negotiation is Completed ------------------

Requested quantity is: 1046 from M2 is answer by R1 with proposed quantity of: 3250
Requested quantity of: 1046 is agreed by:- M2
------------------------------------------ Quantity Negotiation is Completed ---------------
```

Figure 8.5: Agreement and Confirmation

The Figure 8.4 and 8.5 have two different results as output. If two agents are started to communicate and came to an agreement for certain step will not finished as committed negotiation. In other words, every instance of negotiations will not end with confirmation of results. However, regardless of the agent and at least one agent from the pool of agent have committed for a completion of the tasks. According to the Figure 8.5 raw material agent "R1" is committed for a completion of tasks with manufacturer agent "M2". The real comparison depicted that in early stage "M2" have communicated with "R3" but end up with no results. However, sudden "R1" is employed and completed the tasks with "M2". In contrast, the Figure 8.5 elegantly demonstrated that how initially active agent fails in discussion, and final deal has confirmed with the agent who has joined with latter part of the communication.

8.4 Negotiation for Scheduling

Continuous communication resulted seamless schedules in supply chain management. However, agent commitment for better schedules is another smart feature of multi-agent application. Until this stage have been discussed continuous communication for price and quantity, form this stage communication strategy is used for discuss successful scheduling. In this step manufacture agents are communicated with transport agent for the arrangement of transport schedules. There are limited number of transport agents available in this solution and they hold the responsibility for delivering products from manufacture agents to warehouse agent and retailer agents. Sometimes a schedule has tendency to change especially in complex communication structure. Under this chapter have clearly depicted some evaluation points for handling complexity in distributed supply chain environment. Scheduling has three steps in relation to product transport and it is shown in Table 8.2.

Step No	Description
1	The material or product owner is informed to the transport agent about the dates they are willing to dispatch.
2	Transport agent then informed the free slots in their schedule for confirmation.
3	Finally, both parties are agreed on certain date according to the communication.

Table 8.2: Steps in Transport Schedule Communication

There are three figures including Figure 8.6, 8.7 and 8.8 have shown and described later in this chapter to align with the steps given in Table 8.2.

```
No schedule have received from Manufacturer Agent M5 to Transport Agent T2
Willing to Dispatch Products by M1 on or Before ---------------------2015-04-01
Willing to Dispatch Products by M2 on or Before ---------------------2015-04-01
Willing to Dispatch Products by M3 on or Before ---------------------2015-04-01
```

Figure 8.6: Manufacture Agent Product Dispatch Records

Scheduling is sometimes found as difficult, because there are two or three parties have to agree on particular date with some other parameters (quantity, time, method of transport, environmental factors). If one part is not willing to accommodate the services for other parties then the situation became complex. In this research project agents have represented the stakeholders of supply chain with interaction of other agents.

Therefore, negotiation is became imperative in scheduling. The Figure 8.6 illustrated that the manufacturer agents respectively “M1, M2 and M3” are proposed the available dates for to deliver their products to warehouse agents. In prior to an interference of warehouse agent the transport agent had to confirm their availability for delivery schedules. The Figure 8.7 have clearly shown that transport agent delivery schedule for each manufacturer agent. However, manufacturer agent “M5” is willing to dispatch products on 1st April 2015 and it have been confirmed by transport agent “T1”, this scenario have shown in Figure 8.7.

```
No schedule have received from Manufacturer Agent M5 to Transport Agent T2
My (T1) Schedule is: Delivery Start Date:2015-04-01 Delivery End Date: 2015-04-01 Delivery Start Time:
09:00:00 Maximum Quantity: 2000 To M1
----------------------------------------------------------------------------------------------------
No schedule have received from Manufacturer Agent M1 to Transport Agent T1
The agent M3 reply about transport schedule..! to T1
My (T3) Schedule is: Delivery Start Date:2015-04-02 Delivery End Date: 2015-04-02 Delivery Start Time:
08:30:00 Maximum Quantity: 4000 To M3
----------------------------------------------------------------------------------------------------
No schedule have received from Manufacturer Agent M3 to Transport Agent T3
The agent M4 reply Need Schedule to T3
My (T1) Schedule is: Delivery Start Date:2015-04-01 Delivery End Date: 2015-04-01 Delivery Start Time:
09:00:00 Maximum Quantity: 2000 To M3
----------------------------------------------------------------------------------------------------
No schedule have received from Manufacturer Agent M3 to Transport Agent T1
The agent M4 reply about transport schedule..! to T1
My (T3) Schedule is: Delivery Start Date:2015-04-02 Delivery End Date: 2015-04-02 Delivery Start Time:
08:30:00 Maximum Quantity: 4000 To M4
----------------------------------------------------------------------------------------------------
No schedule have received from Manufacturer Agent M4 to Transport Agent T3
The agent M5 reply Need Schedule to T3
My (T1) Schedule is: Delivery Start Date:2015-04-01 Delivery End Date: 2015-04-01 Delivery Start Time:
09:00:00 Maximum Quantity: 2000 To M4
----------------------------------------------------------------------------------------------------
No schedule have received from Manufacturer Agent M4 to Transport Agent T1
The agent M5 reply about transport schedule..! to T1
My (T3) Schedule is: Delivery Start Date:2015-04-02 Delivery End Date: 2015-04-02 Delivery Start Time:
08:30:00 Maximum Quantity: 4000 To M5
----------------------------------------------------------------------------------------------------
No schedule have received from Manufacturer Agent M5 to Transport Agent T3
My (T1) Schedule is: Delivery Start Date:2015-04-01 Delivery End Date: 2015-04-01 Delivery Start Time:
09:00:00 Maximum Quantity: 2000 To M5
```


Figure 8.7: Transport Agent Delivery Schedule

There are various transport agent who has different schedules for manufacture agents. Every manufacture agent have not received same schedule and it has a dynamic structure. In other words, same instance of communication one manufacture agent may receive different schedules form different transport agents. This mechanism provides high availability of schedules for smooth delivery without wasting time and money. According to the Figure 8.7 the active transport agents have different schedules for various manufacturer agents, among others the transport agents “T1” is confirmed the date required by manufacture agent "M5". Therefore the agents "M5 and T1" have confirmed the transport schedule on 1st April 2015. However, the result of the confirmation has been described in Figure 8.8. The circled area represented that confirmation of result and final confirmation message.

```
Willing to Dispatch Products by M1 on or Before ----------------------2015-04-01
Willing to Dispatch Products by M3 on or Before ----------------------2015-04-01
----------- Manufacturer Agent M5 have agreed with Transport Agent T1 on 2015-04-01 ----------- Delivery Confirmed
```

Figure 8.8: Confirmation Message

There are some instance all manufacture agent will not receive schedules from available transport agent. So in this sense, some manufacture agent have to wait considerable time until they will receive their slots from transport agent according to availability of resources. The Figure 8.8 shows that confirmation of transport schedule from both parties. In relation to the Table 8.2 there are two steps have to complete before reaching to the final confirmation stage. The Figure 8.7 and 8.8 have clearly evaluated the outcomes of the schedule negotiation in prior to the final step. Therefore, continuous communication, coordination and negotiation in multi-agent application have committed successful results in scheduling.

8.5 Interference of Human Agent

Emergent property in multi-agent systems have changed the direction of the objectives. In this research project interference of human agent made significant change for the transport schedule according to the change in environment. Agents are proactive and they can exhibit the changes in environment.

Throughout this research project one human agent is initialized to infer and changed the certain schedules have confirmed between transport agent and manufacturer agent. According to the human agent interference the manufacturer and transport agent have changed the objective date to some other date which not appeared before in the schedules of transport agent knowledge base.

```
Received message T1 the content is Hi I am human agent
, please reschedule the delivery confirmed on 2015-04-01,  due to holiday..! and its sent by H1
Received message T2 the content is Hi I am human agent
, please reschedule the delivery confirmed on 2015-04-01,  due to holiday..! and its sent by H1
Received message T3 the content is Hi I am human agent
, please reschedule the delivery confirmed on 2015-04-01,  due to holiday..! and its sent by H1
Send request to M4 for rescheduling transport on 2015-04-02 due to holiday from T1
(INFORM
 :receiver  (set ( agent-identifier :name M4@192.168.10.15:1099/JADE ) )
 :content  "Request new dates...!"
)
Received massage content is... Request new dates...! and it is send by ( agent-identifier :name T1@
Send request to M5 for rescheduling transport on 2015-04-01 due to holiday from T2
(INFORM
 :receiver  (set ( agent-identifier :name M5@192.168.10.15:1099/JADE ) )
 :content  "Request new dates...!"
)
Received massage content is... Request new dates...! and it is send by ( agent-identifier :name T2@
```

Figure 8.9: Human Agent Interference

As discussed in implementation chapter the human agent is represented as “H1”. Human agent is responsible for multicast message to all available transport agent for requesting to change delivery dates according to change in an environment. The Figure 8.9 described, the available and active transport agents have received the message and resend inform message to manufacturer agents. Up to this stage there are confirmed transport schedules in order to start delivery.

With the sudden changed in environment both manufacture agent and transport agent have to reschedule the delivery dates. With the interaction of human agent, the transport agent has delivered the message to manufactures and it is depicted in Figure 8.9 as “request new dates”. This is the situation where initiate an emergent property and it has changed the direction of objective dates. In contract, the details and confirmation of new dates have shown in Figure 8.10.

```
Received massage content is... Request new dates...! and it is send by ( agent-identif
No agreed transport schedule found for T3
Next available dates from manufacturer agent M4 is... 2015-04-02
Next available dates from manufacturer agent M4 is... 2015-04-07
Next available dates from manufacturer agent M4 is... 2015-04-06
Next available dates from manufacturer agent M5 is... 2015-04-02
Next available dates from manufacturer agent M5 is... 2015-04-07
Next available dates from manufacturer agent M5 is... 2015-04-06
The manufacturer agent M4 have agreed on 2015-04-07 to dispatch the products with T1
The manufacturer agent M5 have agreed on 2015-04-02 to dispatch the products with T2
(INFORM
 :receiver  (set ( agent-identifier :name T1@192.168.10.15:1099/JADE  :addresses (sequ
 :content  "New date schedule confirmation is received...!"
 :reply-with  T1@192.168.10.15:1099/JADE1434181591796 )
++++++++++++++++++++++++++++++++++++++++++++++++++ New date schedule negotiation is co
(INFORM
 :receiver  (set ( agent-identifier :name T2@192.168.10.15:1099/JADE  :addresses (sequ
 :content  "New date schedule confirmation is received...!"
 :reply-with  T2@192.168.10.15:1099/JADE1434181591827 )
```

Figure 8.10: Sudden Change in Transport Schedule

According to the "request new dates" message have shown in Figure 8.9 has propagated new direction in transport schedule. The manufactures have received the above mention message are started to resend the next available dates. This incidence has occurred due to certain problem with the early confirmed date of the schedule. In other words, the intended date have changed as a holiday due to some decisions of the government. Therefore, rescheduling is important to avoid any inconsistencies in supply chain functions.

The circled area have mentioned in Figure 8.10 as new reschedule date. With the completion of rescheduled procedure, both agent have informed the new date as confirmation message. In summary, "M5 and T1" have confirmed the delivery date as 1st April 2015 at the beginning of the negotiation. Unexpectedly it has changed and confirmed on 2nd April 2015 due to emergent property in multi-agent application. It is evident that, a multi-agent technology is capable to handle any dynamic changes without affecting the function of the supply chain management.

8.6 Further Negotiation in Supply Chain

There are some remaining stages have to describe during this chapter including communication with retailers and customers. In this stage warehouse agent have joined with the communication process for transport scheduling. This stage evident that all three stakeholders (manufacturers, transport agents and warehouse agents) are confirmed dispatch and receive products according to their schedules. At the beginning of this stage some warehouse agents have received empty schedules. In other words, the confirmation is pending among manufactures and transport agent. With completion of the confirmation among manufacture and transport agent are joined with warehouse agent to complete entire scheduling process. The Figure 8.11 illustrated that completed transport schedule.

```
No confirmed dates received from manufactures and transport agents to warehouse agent W3 gets null
No confirmed dates received from manufactures and transport agents to warehouse agent W3 gets null
No confirmed dates received from manufactures and transport agents to warehouse agent W3 gets null
No confirmation among manufacturer, transport and warehouse agents for agreed date..
Transport agent T2 has agreed with manufacturer agent M5 on the 2015-04-02 is acknowledged by warehouse agent W1

No confirmation among manufacturer, transport and warehouse agents for agreed date..
Transport agent T2 has agreed with manufacturer agent M5 on the 2015-04-02 is acknowledged by warehouse agent W1

No confirmation among manufacturer, transport and warehouse agents for agreed date..
No confirmed dates received from manufactures and transport agents to warehouse agent W2 gets null
No confirmed dates received from manufactures and transport agents to warehouse agent W2 gets null
No confirmed dates received from manufactures and transport agents to warehouse agent W2 gets null
No confirmed dates received from manufactures and transport agents to warehouse agent W3 gets null
No confirmed dates received from manufactures and transport agents to warehouse agent W3 gets null
No confirmed dates received from manufactures and transport agents to warehouse agent W3 gets null
Transport agent T1 has agreed with manufacturer agent M4 on the 2015-04-07 is acknowledged by warehouse agent W1

Manufacture, transport and warehouse agents have agreed on 2015-04-07 for delivery confirmation.................
```


Figure 8.11: Complete Transport Schedule

Supply chain is a continuous process and it activities has certain dependency. Sometimes one activity/function is highly depend on the completion of next activity. Therefore, the retailors have to receive products in order to sell for final stakeholder/customer. However, throughout this chapter have evaluated that including production of raw materials, manufacturing, transport and storing products at warehouse premises.

From this stage onwards the research project have reached the final stage in continuous communication. During this chapter have illustrated not only evaluation strategies in supply chain management but also explained multi-agent features and their dynamics have supported for the successful functioning of the entire business process. Before evaluation of the retailor and customer experience there is another multi-agent feature is have to demonstrate. Butter-fly effect is another multi-agent feature in multi-agent applications. Small change in one end of the application/activity has propagated huge change in other end.

Transport is one of the strategic stage in supply chain management, if any failures in transport stage have been propagated major effect to other activities in SCM. Butter-fly effect has occurred due to unavailability of transport service. The Figure 8.12 has shown distinct features related to butter-fly effect.

```
No confirmed dates received from manufactures and transport agents to warehouse agent W2 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W2 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W2 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W3 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W3 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W3 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W1 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W1 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W1 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W2 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W2 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W2 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W3 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W3 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W3 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W1 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W1 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W1 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W2 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W2 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W2 gets nothing
No confirmed dates received from manufactures and transport agents to warehouse agent W3 gets nothing
```


Figure 8.12: Butter-fly Effect in Transport Scheduling

In some instance active transport agents are not willing to provide the service. This has happened due to certain employee strike or employees are informed as health problems. At the beginning of the deliberation they are active for some instance, but at the end of the communication suddenly decided to prevent the service. Therefore, it leads to collapse the entire scheduling process and as result warehouse agent received nothing during the communication. In general, small modification in end of the transport agent have generated large effect for manufactures and warehouse agents.

8.7 Increased Complexity among Retailer Agents

From this stage on words the complexity in communication have increased drastically due to involvement of larger number of retailers. Retailers are responsible for delivering products to final customers while they received the products from warehouse agents. They propagate multicast message to all available warehouse agents including products types and quantity requirements as a request. This scenario have depicted in Figure 8.13.

```
(INFORM
 :sender  ( agent-identifier :name RE5@192.168.10.15:1099/JADE  :addresses (sequence http://Manoj:7778/acc ))
 :receiver  (set ( agent-identifier :name W1@192.168.10.15:1099/JADE ) )
 :content  "Need product X with quantity 1000"
) ***** RE10
(INFORM
 :sender  ( agent-identifier :name RE1@192.168.10.15:1099/JADE  :addresses (sequence http://Manoj:7778/acc ))
 :receiver  (set ( agent-identifier :name W2@192.168.10.15:1099/JADE ) )
 :content  "Need product Y with quantity 2000"
) ***** RE1
(INFORM
 :sender  ( agent-identifier :name RE2@192.168.10.15:1099/JADE  :addresses (sequence http://Manoj:7778/acc ))
 :receiver  (set ( agent-identifier :name W2@192.168.10.15:1099/JADE ) )
 :content  "Need product Y with quantity 2000"
) ***** RE2
(INFORM
 :sender  ( agent-identifier :name RE3@192.168.10.15:1099/JADE  :addresses (sequence http://Manoj:7778/acc ))
 :receiver  (set ( agent-identifier :name W2@192.168.10.15:1099/JADE ) )
 :content  "Need product Y with quantity 2000"
) ***** RE3
(INFORM
 :sender  ( agent-identifier :name RE4@192.168.10.15:1099/JADE  :addresses (sequence http://Manoj:7778/acc ))
 :receiver  (set ( agent-identifier :name W2@192.168.10.15:1099/JADE ) )
 :content  "Need product Y with quantity 2000"
) ***** RE4
```


Figure 8.13: Retailers are Join with the Negotiation Process

The ACL (Agent Communication Language) message consist of various predefine contents including sender, receiver, content and performative etc. In this stage number of warehouse agent are started to receive requests from retailers of the products and quantity requirements. According to the message space given in Figure 8.13 have depicted the content of requested message with intended receiver. Therefore, each warehouse agent is responsible during this stage to fulfill the retailer requirement with regard to the product availability. If the requested product is not available during this communication cycle, the retailor had to wait considerable time until he/she found the sufficient warehouse agent who has requested products and required quantities. There are some instance all requested retailers are not receive the intended products, in such a situation their requirement will processed to queue.

8.8 Customer Involvement in Communication

SCM is continuous process and its stages are not cascade. In every instance customers are send their requirement to retailers. Retailers have the responsibility to cater customer requirement whenever necessary. There are unlimited customers in nature, but we have stimulated only potion of them.

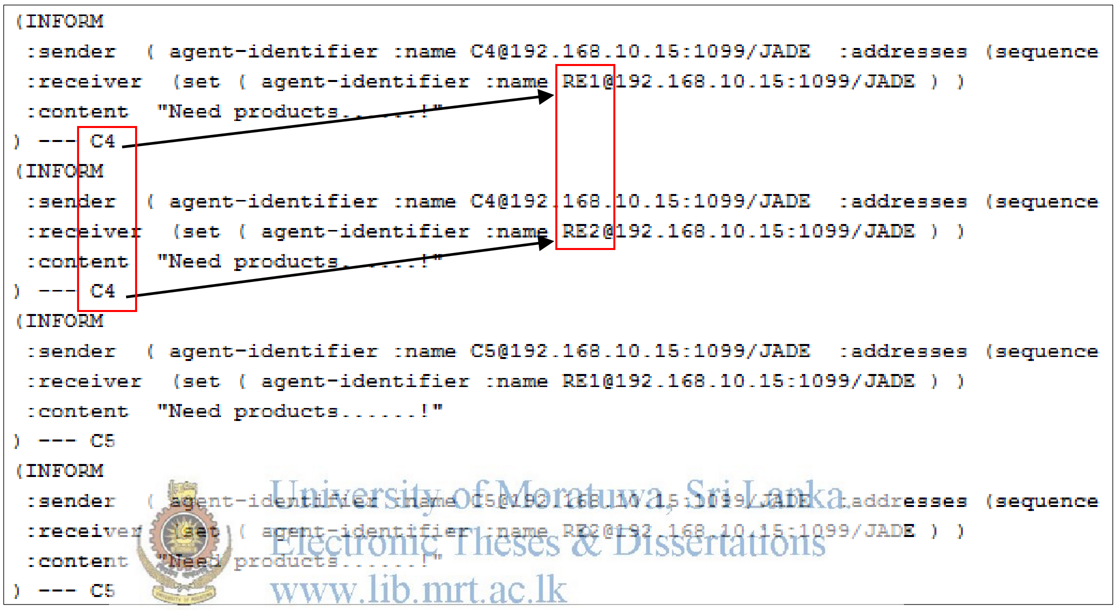

```
(INFORM
 :sender  ( agent-identifier :name C4@192.168.10.15:1099/JADE  :addresses (sequence
 :receiver  (set ( agent-identifier :name RE1@192.168.10.15:1099/JADE ) )
 :content  "Need products......!"
) --- C4
(INFORM
 :sender  ( agent-identifier :name C4@192.168.10.15:1099/JADE  :addresses (sequence
 :receiver  (set ( agent-identifier :name RE2@192.168.10.15:1099/JADE ) )
 :content  "Need products......!"
) --- C4
(INFORM
 :sender  ( agent-identifier :name C5@192.168.10.15:1099/JADE  :addresses (sequence
 :receiver  (set ( agent-identifier :name RE1@192.168.10.15:1099/JADE ) )
 :content  "Need products......!"
) --- C5
(INFORM
 :sender  ( agent-identifier :name C5@192.168.10.15:1099/JADE  :addresses (sequence
 :receiver  (set ( agent-identifier :name RE2@192.168.10.15:1099/JADE ) )
 :content  "Need products......!"
) --- C5
```


Figure 8.14: Customers Demands

According to the Figure 8.14 have illustrated that numerous customer demands for different products. During this communication customer have authority to receive best product among available retailers according to their expectations. In this scenario the customer “C4” is sending same request for retailor “RE1” and “RE2” to receive best product and services from available competition among retailers.

8.9 Simulation and Real Data

This research project not consist of only simulation but also included some real data from the logistics companies available in the market. As discussed early in this chapter, the transport schedule data are received from real scenario. However, the other information during this research project are simulated to describe the functionality in SCM.

8.10 Adoption to Sudden Change in Environment

Throughout this research project agents in multi-agent system application have the capability of adoption to sudden change in the environment. With the competition in the environment there are some agents have received the advantages.

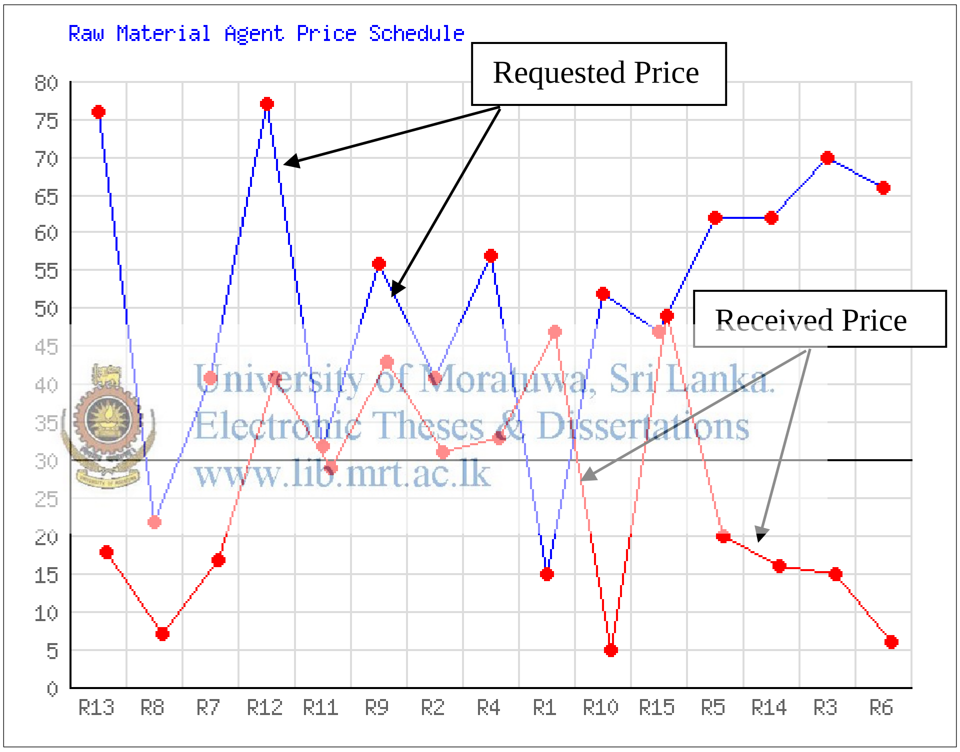


Figure 8.15: Raw Material Agent Requested and Received Price Schedule

At the beginning of the communication, the raw material agents have received relatively high prices, but sudden change in the environment have reduced the received prices drastically. In other words, available numbers of raw material agents are increased in environment have made the different impact for the received prices. The same scenario have depicted in Figure 8.16.

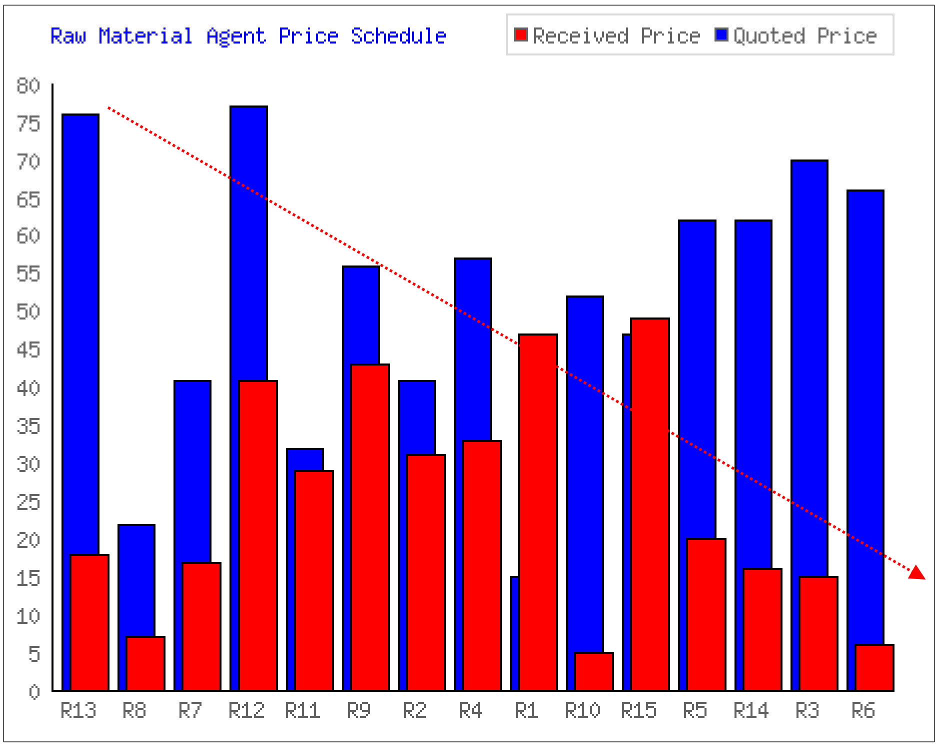


Figure 8.16: Raw Material Agent Requested and Received Price Schedule in Different View

This has the same meaning given if Figure 8.15, but in different angle to describe sudden change in environment. The arrow have drawn on the Figure 8.16 describes sudden recline of the prices due to competition among retailer agents. In SCM the raw material suppliers have the competition among prices, however in some instance they are not willing to supply required quantities. From the pool of suppliers only few of them are able to complete the requirement of manufacturers. This scenario have depicted in Figure 8.17.

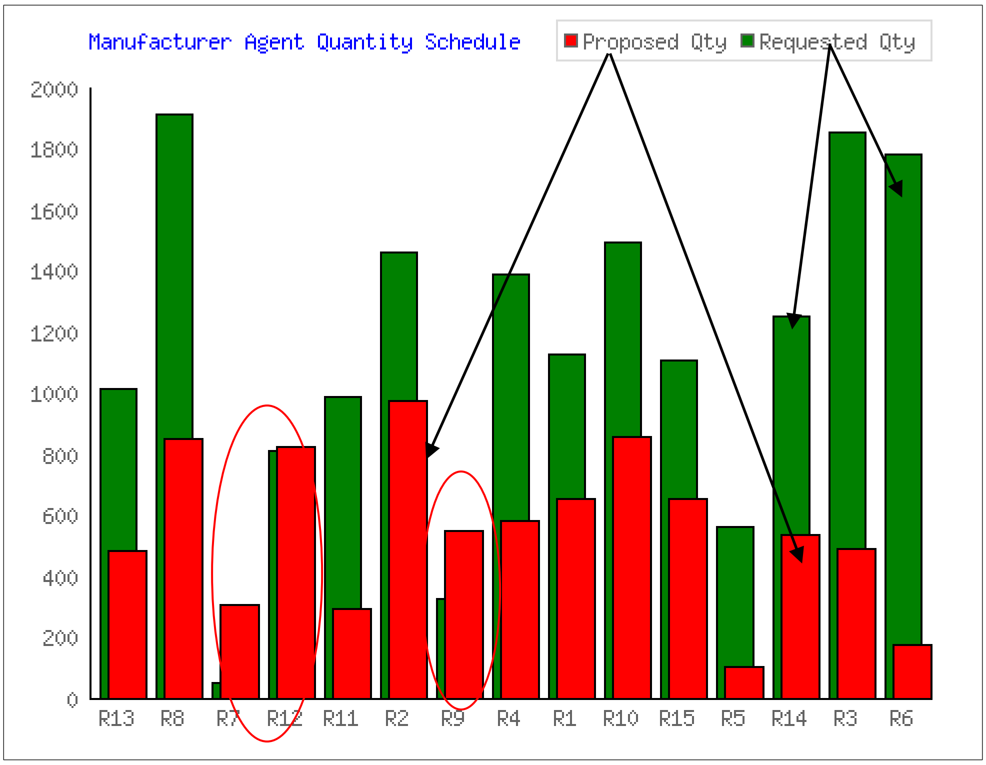


Figure 8.17: Manufacture Agent Proposed and Received Quantities

According to the Figure 8.17 there are few raw material agents have completed the required quantity requirement of manufacturers. As mentioned early in this chapter deliberation among agent have resulted successful outcome in SCM. The ultimate goal of the SCM process is to satisfy the intended stakeholder of the system. During this quantity negotiation there are only three raw material agents have completed required quantity requirements among fifteen suppliers. In general, in one stage a particular agent is performing well, but in subsequent stages some other agent is committed to the completion of the tasks. This is the complexity is associated with SCM applications. However, our MAS solution have concluded the evidence throughout this chapter regardless of the complexity in application to accelerate the tasks completion while satisfying customer demands.

8.11 Summary

In this chapter, we have taken different strategies to conclude the result have obtain from each stage to represent the successfulness of the SCM process. Therefore, the entire evaluation is targeted not only to implement supply chain functions, but also to demonstrate a multi-agent features. This chapter have followed the step by step evaluation process while satisfying the intended stakeholders. The next chapter is about the conclusion and further work of this research project.

Chapter 9

Conclusion & Further Work

9.1 Introduction

The previous chapter illustrated that various evaluation strategies in SCM, and this chapter is mainly focused on the final conclusion and further work. The study consists of systematic review of the limitations in supply chain management and development of complete software component using multi-agent system technology. This research contributes to classification of areas in supply chain management with understanding of restrictions in conventional software systems. Immense uncertainty in complex systems have resulted unpredictable behaviors in nature, however, the importance of decentralized architecture have taken vital position regardless of the application domain. During this application the entire solution have taken the basement as the decentralized architecture.

In the introductory chapter we have explained that introduction, aim and the objectives of the system. Solution in brief with the background and motivation included basic understanding of the system and main drivers to the system.

Large volume of literature review chapter (Current Trends and Issues) described a real world problem and limitation in available technologies. In this stage also included outlining of some industry problems and in what way the concept of multi-agent system technology is address the complexity for solving problems. According to the essentials in multi-agent technology chapter illustrated that various distinct features in multi-agent applications. It also presented that multi agent architecture and motivation to use MAS technology.

With the completion of essentials of multi-agent technology chapter this research project have focused to novel approach for supply chain management. This chapter have described multi agent system technology as novel approach for the supply chain management. In this solution, included that the features, inputs, outputs and behaviors of agent technology, and the power of multi agent concept for solving problems which cannot solve by using conventional supply chain management applications. In addition to that non-functional requirement and users of multi-agent application have identified and described at the end of this chapter.

In design chapter in this research project have focused on two components architecture including information sharing and collaborative planning. It also described essential entities in supply chain environment and their dependency. High-level architecture is included in this chapter and illustrated relationship among entities in SCM. This chapter have taken vital position to describe using design diagrams of agent communication and message passing strategy.

Then the system is focused to shift the design emphasis to implementation. At the beginning of this chapter have introduced agent categorization and their alias of representation. With the description of number of agents join with the communication consisted that complexity of SCM functions. After that the system have implemented common domain ontology and personal ontology to maintain knowledge base of individual agents. Main container and JADE framework facilities have described with the completion of knowledge base while describing agent communication. Also discussed that agent administration is important because agents are entering and leaving the system in distributed environment. Finally, in this chapter described that dynamic report mechanism, hardware and software requirements.

Before starting evaluation chapter this research project have taken important step to describe real world problem using novel approach chapter. In this chapter have illustrated that why SCM is denoted as real world problem and different between conventional system and multi-agent applications.

Finally, this research project have reached to the evaluation chapter where it has evaluated each stage in supply chain against agent features and behaviors. This consisted evidence of early negotiation of agents in supply chain while evaluating their features such as communication, coordination and negotiation. There are significant features have evaluated including emergent property with uncertainty. In other words, interaction of human agent have committed to emergent property with sudden interference in supply chain functions. In addition to that butter-fly effects have experienced during evaluation because of sudden change in transport agent behaviors. In summary, during this chapter have evaluated supply chin outputs and multi-gent features to validate customer requirements.

However, the developed system can be applied to forecast trends in supply chain functions and to maximize business objectives of an organization, and finally satisfy the end customer requirements. In fact, the results of the evaluation chapter have clearly indicates that further negotiation have benefited for the significant change in the SCM which cannot achieve by other conventional technologies. However, we believe that the presented model and multi-agent system technology shows potential for the future researchers to eliminate communication complexity in distributed environment with immense uncertainty.

9.2 Further Work

Multi-agent system technology is suited for any complex problem regardless of the application domain. Throughout this research project different stages in SCM have implemented and evaluated using both real world data and simulation. As a further work the entire functions of the supply chain it to be evaluated using real world data. In addition to that the system is to be deployed in real distributed network to obtain live evidence of communication complexity.

The same methodology of multi-agent system technology is planned to implement in a network system for diagnosing fault and issues arise in organizational network systems. In other words, there is no proper mechanism to detect Distributed Denial of Service Attacks (DDOS Attacks), malfunctioning of nodes and hardware components (Routers, Switchers, Firewalls etc.) in network system. Therefore, as a distinct solution multi-agent system technology is planned to automate the necessary agents to protect organizational network systems.

9.3 Summary

Mainly this chapter have summarized the entire implementation of SCM using multi-agent technology. However, additional steps have taken to test the system using real world data.

References

[1] G. Zhang, B. Eddy Patuwo, and M. Y Hu, “Forecasting with artificial neural networks:: The state of the art,” *Int. J. Forecast.*, vol. 14, no. 1, pp. 35–62, 1998.

[2] “Vidushi Sharma, Sachin Rai and Anurag Dev, 2012, A Comprehensive Study of Artificial Neural Networks, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 10, October 2012.” .

[3] M. Dianati, I. Song, and M. Treiber, “An introduction to genetic algorithms and evolution strategies,” Technical report, University of Waterloo, Ontario, N2L 3G1, Canada, 2002.

[4] E. J. Horvitz, J. S. Breese, and M. Henrion, “Decision theory in expert systems and artificial intelligence,” *Int. J. Approx. Reason.*, vol. 2, no. 3, pp. 247–302, 1988.

[5] M. Ying and Z. Jianyong, “The Study of Multi-agent-based Logistics System of Special Items,” *2nd Int. Conf. Comput. Appl. Syst. Model.*, 2012.

[6] “Engene Santos Jr, Feng Zhang, 2003, Intra Organizational Logistics Management Through Multi Agent System.” .

[7] J. Jorge E. Hernández, Raúl Poler, Mula and D. de La Fuente, “A multiagent based-model for the collaborative planning process in decentralized supply chain networks,” in *XIII Congreso de Ingeniería de Organización*, 2009, pp. 1012–1020.

[8] W. Um, “A Study of Multi-Agent Based Supply Chain Modeling and Management,” *iBusiness*, vol. 02, no. 04, pp. 333–341, 2010.

[9] S. F. Smith and N. M. Sadeh, “A Multiagent Approachi,” *Decis. Sci.*, vol. 29, no. 3, 1998.

[10] “Jayashankar M. Swaminathan, Stephen F. Smith and Norman M. Sadeh, Modeling Supply Chain Dynamics: A Multiagent Approach.” . - 1998 - A Multiagent Approachi.pdf.” .

[11] S. Kumar and U. Kumar, “Java Agent Development Framework,” *Int. J. Res.*, vol. 1, no. 9, pp. 1022–1025, 2014.

[12] “FOUNDATION FOR INTELLIGENT PHYSICAL AGENTS, FIPA 97 Specification, Part 2, Agent Communication Language ‘fipa_OC00018.pdf.’” .

[13] V. Kumar and S. Srinivasan, “A review of supply chain management using multi-agent system,” *Int. J. Comput. Sci. Issues*, vol. 7, no. 5, pp. 198–205, 2010.

[14] “Agent based supply chain management automation,” in *Proceedings of the Eighth International Conference on Computing in Civil and Building Engineering (ICCCBE-VIII)*, 2000, vol. 1006.

[15] M. Ojha, “Optimizing supply chain management using gravitational search algorithm and multi agent system,” in *Proceedings of the International Conference on Soft Computing for Problem Solving (SocProS 2011) December 20-22, 2011*, 2012, pp. 481–491.

[16] X. Li and S. K. Lau, “A multi-agent approach towards collaborative supply chain management,” 2005.

[17] “Thierry Moyaux, Brahim Chaib-draa, and Sophie D’Amours, ‘Supply Chain Management and Multiagent Systems: An Overview’, Springer Abstract, Volume 28, 2006, pp 1-27.” .

[18] J. C. Fransoo and M. J. Wouters, “Measuring the bullwhip effect in the supply chain,” *Supply Chain Manag. Int. J.*, vol. 5, no. 2, pp. 78–89, 2000.

[19] N. Julka, R. Srinivasan, and I. Karimi, "Agent-based supply chain management—1: framework," *Comput. Chem. Eng.*, vol. 26, no. 12, pp. 1755–1769, 2002.
[20] M. d'Inverno, D. Kinny, M. Luck, and M. Wooldridge, "A formal specification of dMARS," in *Intelligent Agents IV Agent Theories, Architectures, and Languages*, Springer, 1998, pp. 155–176.
[21] D. Pereira, E. Oliveira, N. Moreira, and L. Sarmento, "Towards an architecture for emotional BDI agents," in *EPIA*, 2005, vol. 5, pp. 40–47.
[22] X. Zhu, "Agent Based Modeling for Supply Chain Management: Examining the Impact of Information Sharing," Kent State University, 2008.
[23] R. Carvalho and L. Custódio, "A multiagent systems approach for managing supply-chain problems: new tools and results," *Intel. Artif.*, vol. 9, no. 25, pp. 79–88, 2005.
[24] C. I. Popirlan, A. Stefãnescu, and L. Stefãnescu, "Multi-Agent Approach for Data Analysis in a Knowledge-based System for Contact Centers," *World Acad. Sci. Eng. Technol.*, vol. 59, pp. 1126–1131, 2011.
[25] A. K. Chopra and M. P. Singh, "Elements of a business-level architecture for multiagent systems," in *Programming Multi-Agent Systems*, Springer, 2010, pp. 15–30.
[26] S. Srinivasan, S. K. Singh, and V. Kumar, "Multi-agent System based Service Oriented Architecture for Supply Chain Management System," *Int. J. Comput. Appl.*, vol. 27, no. 5, pp. 12–16, 2011.
[27] Haitham Al-zu'bi, (2010), Applying Electronic Supply Chain Management Using Multi-Agent System: A Managerial Perspective,International Arab Journal of e-Technology, Vol. 1, No. 3,.
[28] Y. Fu, R. Piplani, R. De Souza, and J. Wu, "Multi-agent enabled modeling and simulation towards collaborative inventory management in supply chains," in *Simulation Conference, 2000. Proceedings. Winter*, 2000, vol. 2, pp. 1763–1771.
[29] H. A. Rady, "Multi-Agent System for Negotiation in a Collaborative Supply Chain Management.," *Int. J. Video Image Process. Netw. Secur.*, vol. 11, no. 5, 2011.
[30] Y. Chen, Y. Peng, T. Finin, Y. Labrou, S. Cost, B. Chu, R. Sun, and R. Willhelm, "A negotiation-based multi-agent system for supply chain management," *Work. Notes Agents*, vol. 99, 1999.
[31] B. Hettige, A. S. Karunananda, and G. Rzevski, (2012), "Multi-agent System Technology for Morphological Analysis",Sri Lnka Association of Artificial Intelligence, (SLAAI).
[32] K. Psarris and World Scientific and Engineering Academy and Society, Eds., *Proceedings of the 6th WSEAS International Conference on Telecommunications and Informatics (TELE-INFO '07), 6th WSEAS International Conference on Signal Processing (SIP '07): Dallas, Texas, USA, March 22 - 24, 2007*. Athens: WSEAS Press, 2007.
[33] F. Wu, S. Zilberstein, and X. Chen, "Online planning for multi-agent systems with bounded communication," *Artif. Intell.*, vol. 175, no. 2, pp. 487–511, Feb. 2011.
[34] M. de Weerdt, A. ter Mors, and C. Witteveen, "Multi-agent Planning," *Dept Softw. Technol. Delft Univ. Technol.*, 2003.
[35] I. J. Chen and A. Paulraj, "Understanding supply chain management: critical research and a theoretical framework," *Int. J. Prod. Res.*, vol. 42, no. 1, pp. 131–163, Jan. 2004.

[36] G. P. Kurien and M. N. Qureshi, “Study of performance measurement practices in supply chain management,” *Int. J. Bus. Manag. Soc. Sci.*, vol. 2, no. 4, pp. 19–34, 2011.
[37] R. P. Kampstra, J. Ashayeri, and J. L. Gattorna, “Realities of supply chain collaboration,” *Int. J. Logist. Manag.*, vol. 17, no. 3, pp. 312–330, 2006.
[38] “George Rzevski, 08 New Directions in Searching for Artificial Intelligence.pdf.” .
[39] N. Fabbe-Costes and M. Jahre, “Supply chain integration and performance: a review of the evidence,” *Int. J. Logist. Manag.*, vol. 19, no. 2, pp. 130–154, Aug. 2008.
[40] R. H. Ballou, “Business logistics: importance and some research opportunities,” *Gest. Produção*, vol. 4, no. 2, pp. 117–129, 1997.
[41] T. Blecker and Hamburg International Conference of Logistics, Eds., *Innovative methods in logistics and supply chain management: current issues and emerging practices*, 1. ed. Berlin: epubli, 2014.
[42] J. Storey, C. Emberson, J. Godsell, and A. Harrison, “Supply chain management: theory, practice and future challenges,” *Int. J. Oper. Prod. Manag.*, vol. 26, no. 7, pp. 754–774, Jul. 2006.
[43] “Eugénio Oliveira, Klaus Fischer and Olga Stepankova, 1999, Multi-agent Systems: Which Research for which Applications.” .
[44] F. L. Bellifemine, *Developing multi-agent systems with jade*. Hoboken, NJ: John Wiley, 2007.
[45] O. M. Shehory, *Architectural properties of multi-agent systems*. Carnegie Mellon University, The Robotics Institute, 1998.
[46] “Amit K. Chopra and Munindar P. Singh, ‘Agent Communication’, MIT Press, 2011, www.csc.ncsu.edu/faculty/mpsingh/papers/mas/Agent-Communication-chapter.pdf.” .

[47] “Anand S. Rao and Michael P. Georgef, ‘BDI Agents: From Theory to Practisce’, Aprial 1995.” .
[48] B. Anandampilai, “Content-Based Multicasting Using JADE,” *Int. J. Soft Comput.*, vol. 2, no. 3, pp. 422–425, 2007.
[49] “NICHOLAS R. JENNINGS, MICHAEL WOOLDRIDGE, KATIA SYCARA, A Roadmap of Agent Research and Development, 1998 ‘roadmap.pdf.’” .
[50] K. P. Sycara, “Multiagent systems,” *AI Mag.*, vol. 19, no. 2, p. 79, 1998.
[51] B. Marzougui and K. Barkaoui, “Interaction Protocols in Multi-Agent Systems based on Agent Petri Nets Model,” *Interaction*, vol. 4, no. 7, 2013.

Appendix A

Essentials of Multi Agent technology

A.1 Introduction

This appendix consists of the features in multi-agent systems and also describes the difference between complex system and others.

CLASSES/ Features	RANDOM SYSTEMS	COMPLEX SYSTEMS	STABLE SYSTEMS	ALGORITHMIC SYSTEMS
Predictability	Total uncertainty	Considerable uncertainty	No uncertainty	No uncertainty
Behavior	Random	Emergent	Planned	Deterministic
Norms of behavior	Total freedom of behavior	Some external guidance is essential	Governed by laws and regulations	Follows instructions
Degree of organization	None	Self-organization	Organized	Rigidly structured
Degree of control	None	Self-control by self-organization	Centralized control	No need for control
Irreversible changes	Random changes	Co-evolves with environment	Small temporary deviations possible	None
Operating point	None	Operates far from equilibrium	Operates at an equilibrium	Operates according to the specification

Table A.1: A Multi Agent System Features

Source: George Rzevski, a New Direction of Research into Artificial Intelligence

Appendix B

Design of MAS for SCM

B.1 Introduction

As said in design chapter, this appendix consist of various design diagrams to align with implementation stage.

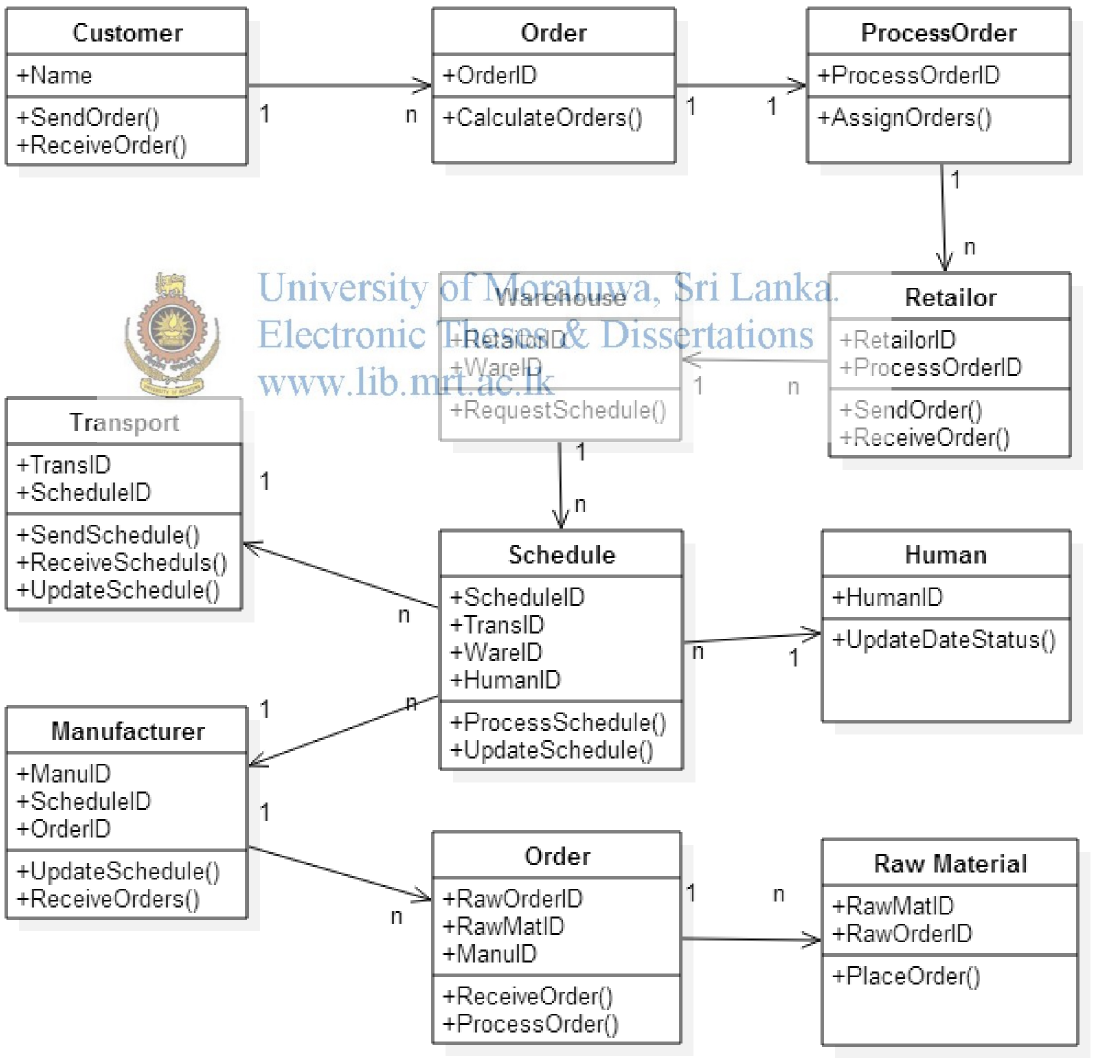


Figure B.1: Class Diagram for SCM

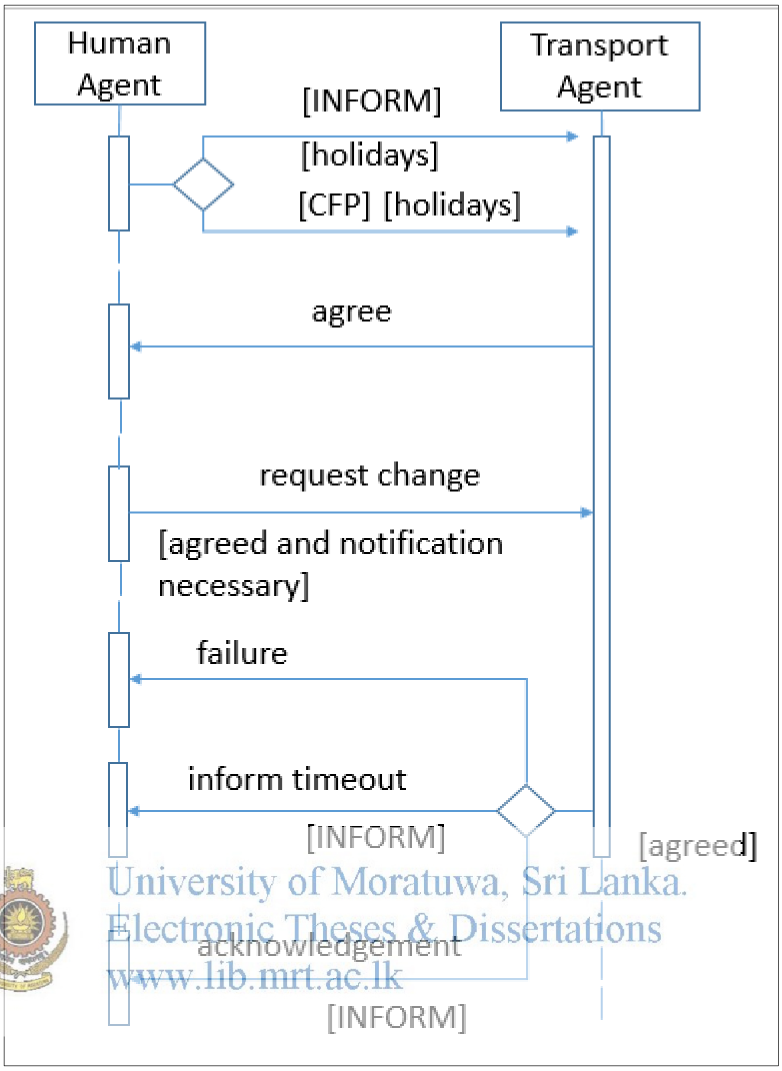


Figure B.2: Sequence Diagram for Human Agent Interaction

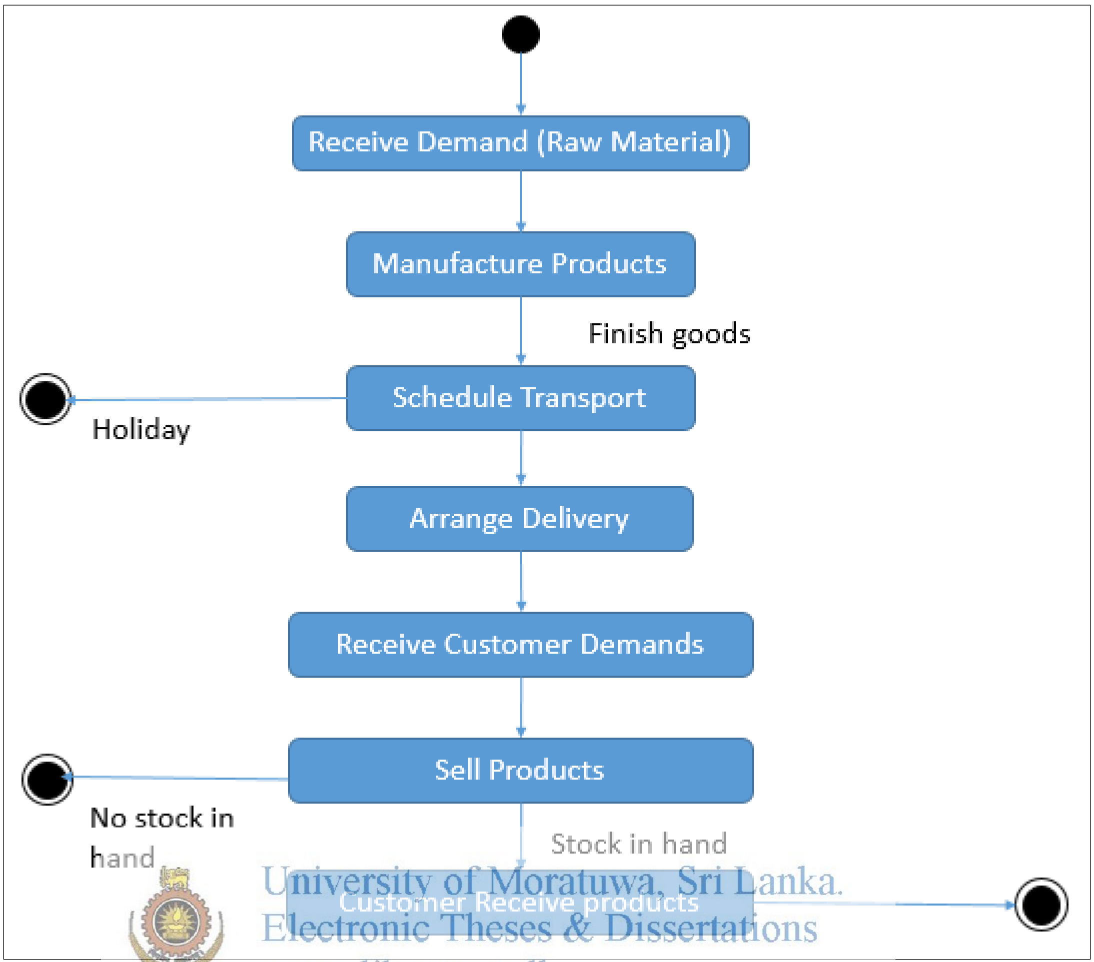


Figure B.3: State Transition Diagram

Appendix C

Implementation of the MAS Solution for SCM

C.1 Introduction

During this appendix have included important code segments, figures and data have used in implementation stage.

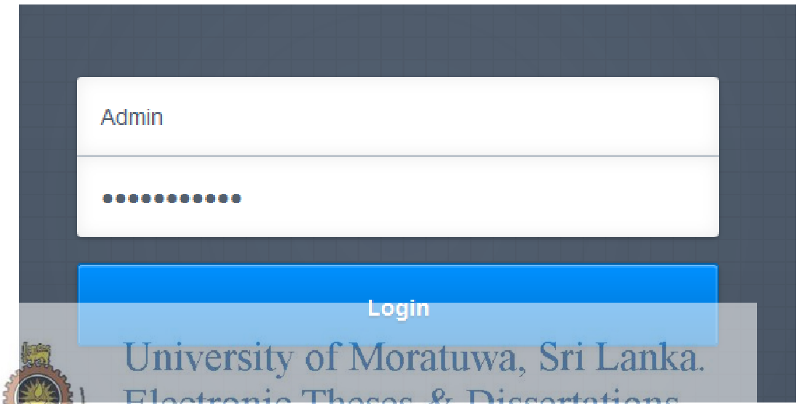


Figure C.1: Login Screen of MASSCM

Figure C.2: Dashboard of MASSCM

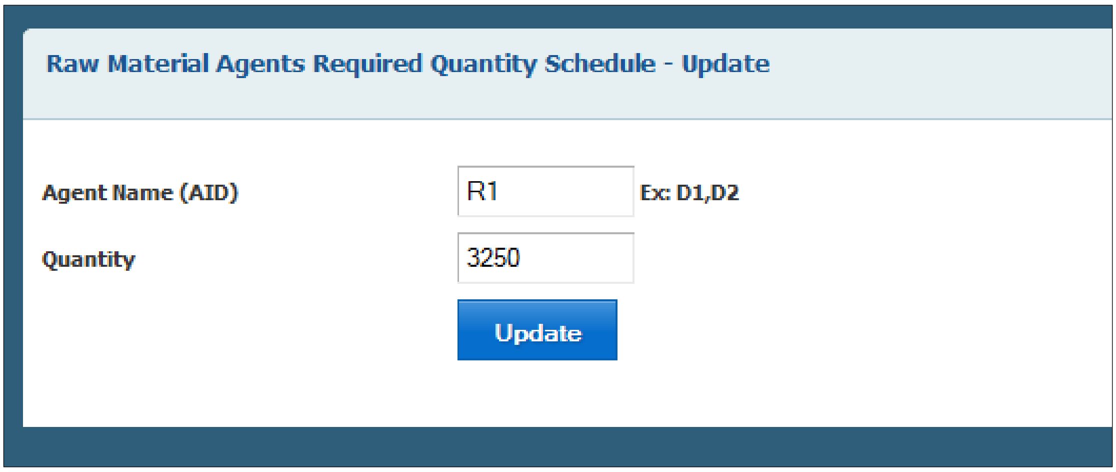


Figure C.3: Ontology Update form

schedule_id	manu_aid	tr_aid	confirm_date	confirm_status
1	M5	T2	2015-04-01	1
2	M5	T2	2015-04-01	1
3	M5	T2	2015-04-01	1
4	M5	T2	2015-04-01	1
5	M5	T2	2015-04-01	1
6	M5	T2	2015-04-01	1
7	M5	T2	2015-04-01	1
8	M5	T2	2015-04-01	1
9	M5	T2	2015-04-01	1
10	M5	T2	2015-04-01	1

Figure C.4: Transport Schedule Data

Source: Link Natural Products (PVT) LTD

tr_autu_id	tr_aid	tr_sdate	tr_edate	tr_time	tr_qty	read_status	manu_aid	tr_date_holiday
1	T1	2015-04-01	2015-04-01	09:00:00	2000	1	M3	1
2	T2	2015-04-01	2015-04-02	12:00:00	300	1	M5	0
3	T3	2015-04-02	2015-04-02	08:30:00	4000	1	M2	0
4	T1	2015-04-02	2015-04-02	06:30:00	200	1	M3	0
5	T3	2015-04-01	2015-04-01	14:30:00	250	1	M2	0
6	T2	2015-04-02	2015-04-02	23:00:00	500	1	M5	0
7	T3	2015-04-01	2015-04-01	12:00:00	700	1	M2	0
8	T1	2015-04-07	2015-04-08	06:30:00	500	1	M3	0
9	T3	2015-04-06	2015-04-07	12:00:00	300	1	M2	0
10	T2	2015-04-02	2015-04-02	06:30:00	2000	1	M5	0

Figure C.5: Transport Schedule Data (Confirmed by Manufacturers)

tr_manu_autoid	tr_manu_aid	tr_manu_date	tr_manu_confirm
1	M1	2015-04-01	0
2	M2	2015-04-06	0
3	M3	2015-04-02	0
4	M4	2015-04-02	0
5	M5	2015-04-01	0

Figure C.6: Transport Schedule of Manufacturers Agents

```
public class DataAccess {

    Statement stmt = null;

        // Select SQL query.

        public ResultSet SelectSql(String query) {

                ResultSet rs = null;
                Statement stmt = null;

                try {

                        DbCon useDbCon = new DbCon(); // Creating instance for a DbCon class.
                        Connection newCon = useDbCon.conn; // Accessing variable which used in DbCon class.

                        stmt = newCon.createStatement();
                        rs = stmt.executeQuery(query);

                        }
                            catch (SQLException ex){
                            ex.printStackTrace();
                        }
```

Figure C.7: Code for MYSQL Data Base Connection

```java
private ACLMessage msg;

public ACLMessage getMessage() {
    return msg;
}

public myReceiver(Agent a, int millis, MessageTemplate mt) {
    super(a);
    timeOut = millis;
    template = mt;
}

public void onStart() {
    wakeupTime = (timeOut < 0 ? Long.MAX_VALUE
            : System.currentTimeMillis() + timeOut);
}

public boolean done() {
    return finished;
}

public void action() {
    if (template == null) {
        msg = myAgent.receive();
    } else {
        msg = myAgent.receive(template);
    }
```


Figure C.8: Message Space Agent Code

```java
// T - Transport

myAgent.getContainerController().createNewAgent("T1","pk1.TransportAgent",null).start();
myAgent.getContainerController().createNewAgent("T2","pk1.TransportAgent",null).start();
myAgent.getContainerController().createNewAgent("T3","pk1.TransportAgent",null).start();

// W - Warehouse

myAgent.getContainerController().createNewAgent("W1","pk1.WarehouseAgent",null).start();
myAgent.getContainerController().createNewAgent("W2","pk1.WarehouseAgent",null).start();
myAgent.getContainerController().createNewAgent("W3","pk1.WarehouseAgent",null).start();

// H - Human Agent

myAgent.getContainerController().createNewAgent("H1","pk1.HumanAgent",null).start();
```

Figure C.9: Agent Initialization Code

```java
///Access XML content
try {

    //File ontoXmlFile = new File("C://Users//Manoj Lap//Documents//NetBeansProjects//MAS_SCM
    File ontoXmlFile = new File("C://Users//Manoj Lap//Documents//NetBeansProjects//MAS_SCM//
    JAXBContext jaxbContext = JAXBContext.newInstance(pk2.ontology.AgentConfiguration.class);

    Unmarshaller jaxbUnmarshaller = jaxbContext.createUnmarshaller();
    mySelf = (AgentConfiguration) jaxbUnmarshaller.unmarshal(ontoXmlFile);
```

Figure C.10: Common Domain Ontology Access (XML Access)

```java
for (int i = 0; i < agents.length; i++) {

    msg.addReceiver(/*new AID("M" + i, AID.ISLOCALNAME)*/agents[i].getName());
    par.addSubBehaviour(new myReceiver(this, 2000, template) {
        public void handle(ACLMessage msg) {
                                        // Ontology should goes here...........

            if (msg != null) {
                int offer = Integer.parseInt(msg.getContent());

                System.out.println("Got quote Rs:" + offer
                        + " from " + msg.getSender().getLocalName());
                if (offer <= bestPrice) {
                    bestPrice = offer;
                    bestOffer = msg;
                }
            }
        }
    });
}
```


Figure C.11: Sending Multicast Messages to Manufacturer Agents

```java
seq.addSubBehaviour(new myReceiver(this, 2000,
        MessageTemplate.and(
                MessageTemplate.MatchConversationId(msg.getConversationId()),
                MessageTemplate.or(
                        MessageTemplate.MatchPerformative(ACLMessage.AGREE),
                        MessageTemplate.MatchPerformative(ACLMessage.REFUSE)))) {

            private static final long serialVersionUID = 1L;
```

Figure C.12: Agent Performatives Code

```
seq.addSubBehaviour(new DelayBehaviour(this, rnd.nextInt(3000)) {
    public void handleElapsedTimeout() {
        if (bestOffer == null) {
            System.out.println("Got no quotes");
        } else {
                            // Inser query.............

            String query = "INSERT INTO agents_manufacture"
                    + "(mf_best_price, mf_aid, mf_conid) VALUES "
                    + "('" + bestPrice + "','" + bestOffer.getSender().getLocalName() +
            try {
                da.InsertSql(query);
            } catch (SQLException ex) {
                ex.printStackTrace();
            }

            System.out.println("\nBest Price Rs:" + bestPrice
                    + " from " + bestOffer.getSender().getLocalName());
            ACLMessage reply = bestOffer.createReply();
                                        // Ontology should goes here.......................
            if (bestPrice <= 50) {
                reply.setPerformative(ACLMessage.REQUEST);
                reply.setContent("" + rnd.nextInt(80));
                System.out.print("ORDER at Rs:" + reply.getContent() + " ");
                send(reply);
            }
        }
```

Figure C.13: Ontology Update and Send Reply to Manufacturers


```
addBehaviour(new MessageReceivingBehaviour());
// This code need to modify.........................................
MessageTemplate mt1 =
        MessageTemplate.and(
            MessageTemplate.MatchPerformative( ACLMessage.INFORM ),
            MessageTemplate.MatchSender( new AID( "T1",
                                    AID.ISLOCALNAME))) ;

MessageTemplate mt2 =
        MessageTemplate.and(
            MessageTemplate.MatchPerformative( ACLMessage.INFORM ),
            MessageTemplate.MatchSender( new AID( "T2",
                                    AID.ISLOCALNAME))) ;
MessageTemplate mt3 =
        MessageTemplate.and(
            MessageTemplate.MatchPerformative( ACLMessage.INFORM ),
            MessageTemplate.MatchSender( new AID( "T3",
                                    AID.ISLOCALNAME))) ;
```

Figure C.14: Message Template for to Receive Messages from Transport Agent

```
protected void onTick() {

    ACLMessage receiveFromT1 = receive(mtT1);
    if(receiveFromT1 != null){
        try {
            String maid1 = "";
            String traid1= "";
            Date agDate1 = null;
            try {
                //System.out.println(receiveFromT1.getContent()+" ||| "+receiveFromT1.getSend
                String sql1 = "SELECT DISTINCT manu_aid,tr_aid,confirm_date FROM agents_deliv
                rs = da.SelectSql(sql1);
                while(rs.next()){
                    maid1 = rs.getString("manu_aid");
                    traid1 = rs.getString("tr_aid");
                    agDate1 = rs.getDate("confirm_date");
                }   } catch (SQLException ex) {
                    ex.printStackTrace();
                }

            // print
            System.out.println("Transport agent "+traid1+" has agreed with manufacturer agent

            // Arrange ware date and trans date are equals

            String conSql1 = "SELECT ware_aid FROM agents_delivery_ware WHERE ware_aid='"+myA
            rs = da.SelectSql(conSql1);
           boolean val1 = rs.next();
            if(val1 == false){
            System.out.println("No confirmation among manufacturer, transport and warehouse a
```

Figure C.15: Warehouse Agent Ontology Management


```
String genCID() {
    if (cidBase == null) {
        cidBase = getLocalName() + hashCode()
                + System.currentTimeMillis() % 10000 + "_";
    }
    return cidBase + (cidCnt++);
}

ACLMessage newMsg(int perf, String content, AID dest) {
    ACLMessage msg = newMsg(perf);
    if (dest != null) {
        msg.addReceiver(dest);
    }
    msg.setContent(content);
    return msg;
}
```

Figure C.16: Warehouse Agent Utility Methods

```java
@Override
protected void setup() { // Setup start
    super.setup(); //To change body of generated methods, choose Tools | Template
    addBehaviour(new TickerBehavior(this, 20000));
    addBehaviour(new delayBehaviorOnTrans (this, 30000));



     SequentialBehaviour seq = new SequentialBehaviour();
     addBehaviour(seq);

     ParallelBehaviour par = new ParallelBehaviour(ParallelBehaviour.WHEN_ALL);
     seq.addSubBehaviour(par);


     // This behavior is created to send agreed dates from transport agent and ma

     seq.addSubBehaviour(new DelayBehaviour(this, 90000){
```

Figure C.17: Transport Agent Different Behaviors

```java
// Creat data base connection to obtain holidays..........

query = "SELECT DISTINCT tr_sdate FROM agents_transport_schedule_manu WHERE tr_date_holiday='1'";
String sdata = "";
rs = da.SelectSql(query);
    try {
        while(rs.next()){
            holiday = rs.getDate("tr_sdate");

            String dateUpdate = "INSERT INTO agents_human_holiday (haid,h_holi_dates) VALUES "
                    + "('"+myAgent.getLocalName()+"','"+holiday+"')";
            da.InsertSql(dateUpdate);

            sdata = sdata+holiday+", ";

        }   } catch (SQLException ex) {
        ex.printStackTrace();
    }

    msg.setContent("Hi I am human agent\n, please reschedule the delivery confirmed on "+sdata+" due

    //System.out.println(sdata);
    for(int i = 1; i<=3; i++){
    msg.addReceiver(new AID("T"+i, AID.ISLOCALNAME));
    send(msg);
    removeBehaviour(this);
}
```


Figure C.18: Human Agent Interference

```
// To w1

    ACLMessage toW1 = new ACLMessage(ACLMessage.INFORM);
    toW1.setContent("Need product X with quantity 1000");
    toW1.addReceiver(new AID("W1", AID.ISLOCALNAME));
    send(toW1);
    System.out.println(toW1+" --- "+toW1.getSender().getLocalName());
// To w2

    ACLMessage toW2 = new ACLMessage(ACLMessage.INFORM);
    toW2.setContent("Need product Y with quantity 2000");
    toW2.addReceiver(new AID("W2", AID.ISLOCALNAME));
    send(toW2);
    System.out.println(toW2+" --- "+toW2.getSender().getLocalName());
// To w3

    ACLMessage toW3 = new ACLMessage(ACLMessage.INFORM);
    toW3.setContent("Need product Z with quantity 3000");
    toW3.addReceiver(new AID("W3", AID.ISLOCALNAME));
    send(toW3);
    System.out.println(toW3+" --- "+toW3.getSender().getLocalName());
}
```

Figure C.19: Retailor and Warehouse Agents Negotiation


```
ParallelBehaviour par = new ParallelBehaviour(ParallelBehaviour.WHEN_ALL);
seq.addSubBehaviour(par);

seq.addSubBehaviour(new DelayBehaviour(this, 100050){
public void handleElapsedTimeout(){

    // To w1

    ACLMessage toRe1 = new ACLMessage(ACLMessage.INFORM);
    toRe1.setContent("Need products......!");
    toRe1.addReceiver(new AID("RE1", AID.ISLOCALNAME));
    send(toRe1);
    System.out.println(toRe1+" --- "+toRe1.getSender().getLocalName());

    // To w2

    ACLMessage toRe2 = new ACLMessage(ACLMessage.INFORM);
    toRe2.setContent("Need products......!");
    toRe2.addReceiver(new AID("RE2", AID.ISLOCALNAME));
    send(toRe2);
    System.out.println(toRe2+" --- "+toRe2.getSender().getLocalName());

}
});
```

Figure C.20: Customer and Retailer Negotiation