

# WMAC: Web-Based Multi-Agent Solution for Agriculture Community

H. M. H. R. Jayarathna<sup>1</sup> and B. Hettige<sup>2</sup>

<sup>1</sup>Faculty of Information Technology, University of Moratuwa, Sri Lanka

<sup>2</sup>Department of Statistics and Computer Science, University of Sri Jayawardenapura, Sri Lanka  
harshanajayarathna@gmail.com<sup>1</sup>, buddhitha@dscs.sjp.ac.lk<sup>2</sup>

**Abstract - Multi-agent System Technology (MAS) is one of the powerful technologies which is used to solve real world problems. Compared with existing Multi-agent applications, web based solutions are more useful than the standalone applications. This paper presents web based multi agent system named WMAC which can be used to communicate with the people engaged in Agriculture industry. Web based multi-agent system uses a common MYSQL database as the ontology of each agent. Agents work as web clients and design through the PHP and AJAX technologies. System provides four types of agents as farmer, buyer, seller and instructor; which represent farmers, buyers, sellers and technical instructors in the agricultural community and which makes the communication among persons in the agricultural industry as required. The WMAC is a web based development of existing Java based standalone multi-agent system named AgriCom. The WMAC system has been successfully tested with the practical environment and successful results were achieved.**

## 1. INTRODUCTION

In traditional artificial intelligent systems such as Expert Systems, Genetic Algorithms, Artificial Neural Networks, Fuzzy Logic and Decision Trees it is considered that intelligence can be achieved via stored knowledge. Multi-Agent System (MAS) technology is the novel concept in the artificial intelligent field and this system does not contain pre stored knowledgebase as intelligence. Though MAS technology is considered as a sub area of Artificial intelligence (AI), the MAS technology is beyond the field of AI.

MAS software are having some advantages compared to conventional software such as being Distributed, Knowledge guided, messaging capability, emergent behavior, adaptive, self-organizing and evolving. In multi agent technology a large number of small programs are used which are called agents. Those small programs are run simultaneously and they to communicate among themselves to find suitable answers for problems.

Multi-Agent Engine, Virtual World, Ontology and Interfaces are the common components which are available in multi agent systems [1]. At

first the Multi-Agent Engine starts and gives support at the run time of the system. The environment of the Multi agent system is provided by Virtual world. Agents in Virtual world compete and cooperate together. Knowledge of the each agent of conceived problem domain is available in Ontology. The interface provides support to communicate with the user and the system.

At present web technology is a very powerful information distributed media among people around the world, because anyone can access to websites from any place where the computer and internet connection are available. To access websites, the only requirement is web browser and no need for any other software to be installed. But the use of a standalone application is somewhat difficult, because it may need to install some software such as java runtime environment, .Net framework, etc... Therefore development of standalone applications using java, Visual basic or any other languages is reduced to the field of information distribution.

This paper presents a web based multi-agent system named WMAC which can be used to communicate with people in the Agriculture industry. WMAC system provides four types of agents as farmer; buyer, seller and instructor which represent farmers, buyers, sellers and technical instructors in the agricultural community. System provides the facility to communicate among persons in the agricultural industry when required.

The rest of the paper is organized as follows. Section 2 gives a brief introduction about multi-agent systems including existing multi-agent systems. Section 3 presents design of the WMAC system including a brief description of each module. Then section 4 gives how system works for the practical environment. Finally section 5 gives conclusion and further works for the project.

## 2. MULTI-AGENT TECHNOLOGY

Multi-Agent System Technology is a novel concept in software development field. This technology is very effective than conventional software development due to number of advantages. Multi-agent engine, Virtual world, Ontology and interfaces are the common components which are available in multi agent systems [1]. At first the Multi-agent engine starts and gives the support to run time of system. The environment of the Multi agent system is provided by Virtual world. Agents in Virtual world compete and cooperate together. Knowledge of the each agent of conceived problem domain is available in Ontology. The Interface provides support to communicate with the user and the system.

At present number of toolkits and frameworks are available to develop MAS software. Most of them are built up with Java programming language and few are built up with C/ C++ programming language.

Among them; JADE [10], Jason [11], AgentBuilder [12], ZEUS [13], NetLogo [14], Madkit [15], FIPA-OS [16], AgentSheets [17], Jack [18] and OpenCybele [19] are popular.

JADE (Java Agent DEvelopment Framework) is more than a development framework, which can be considered as a platform. In fact JADE can be distributed over several hosts rather than hosting on a single machine. Agents can also migrate or clone themselves to other hosts of the platform, regardless of the OS. JADE runs on Java virtual machine.

Jason is an Open Source fully java based Multi agent development framework developed through the improved version of agent oriented language AgentSpeak. Through the Jason, developers can easily programmed agent behavior of individual agents.

AgentBuilder is an integrated software development tool to develop general purpose multi-agent systems that allows software developers with no intelligent agent technologies to quickly and easily build intelligent agent-based applications. To develop multi-agent systems, researchers used standard framework or their own agent development frameworks. Further, there are number of agricultural systems which have been designed using Multi-agent System technology.

ZEUS is an Open Source agent development framework, which is also implemented in Java. This framework is also complying with FIPA specifications. ZEUS is famous as an agent development framework, which has the facility to

plan and schedule the actions by agents. It has been specifically mentioned that ZEUS is ideal for the development of collaborative multi agent systems.

MaDKit is an open source modular and scalable multi-agent platform written in Java and built upon the AGR (Agent/Group/Role) organizational model. MaDKit agents play roles in groups and thus create artificial societies.

FIPA-OS (FIPA Open Source) is an open agent platform originating from Nortel Networks. The platform supports communication between multiple agents using an agent communication language which conforms to the FIPA (Foundation for Intelligent Physical Agents) agent standards. A key focus of the platform is that it supports openness. This is naturally supported by the agent paradigm itself and by the design of the platform itself whose parts have loose coupling such that extensions and innovations to support agent communication can occur in several key areas. FIPA-OS is being deployed in several domains including virtual private network provisioning, distributed meeting scheduling and a virtual home environment. It has been demonstrated to interoperate with other heterogeneous FIPA compliant platforms and is in use in numerous institutions around the world.

AgentSheets is an agent-based simulation tool that lets end-users create simulations within an easy to use GUI development environment. This implements the society through a combination of Java authoring tools, spreadsheets and agents. Methods, rules, conditions, and actions are all predefined structures within a GUI palette and they are drag-and dropped into agents by the user. The user is responsible for filling in the corresponding parameters. Tactile programming is the concept which is used by the AgentSheets. In this model agents are visually developed and must be graphically represented within their environment [31]. The emphasis is placed on visual development and the communication of the underlying rules, behaviors, and actions of agents during real-time simulation. The programming becomes tactile because of the heavy emphasis on GUI's for developing and manipulating agents.

Jack is a development environment that is built on top of java and acts as an extension of Java that offers classes for implementing agent behavior.

In OpenCybele, Agents are defined as “a group of event-driven activities that share data, thread, and execution concurrency structure”. Here Activities are “active objects” internal to the agent and act on internal data in response to incoming events. The fact that other agents are

incapable of manipulating the internal data of another reinforces the notion of agent autonomy.

### 3. DEVELOPED SYSTEMS VIA MAS TECHNOLOGY

The multi agent intervention to web technology is low compared to others. However there are number of systems but not for the web technology. Those systems have been developed using Java, C, LIPS and etc.,

iJADE Reporter has been developed by Eddie C.L. Chan and Raymond S.T. Lee [2]. iJADE Reporter is an intelligent context-aware news reporting system. In this, according to the multi-agent architecture news content are categorized by information retrieval algorithm. Frequency with Inverse Document Frequency (TFIDF) is used to separate the news content. This system provides simple, fast and efficient query in internet.

Catholijin M. Jonker, Remco A. Lam and Jan Treur proposed a Multi-agent Architecture for an intelligent website in insurance [3]. The agents in this architecture are based on generic broker agent model.

Przemyslaw Kazienko has developed a Multi-agent system for web advertising [4]. The main function of this system is to provide advertisements, which are most suitable for the given anonymous user navigating the web site.

Matsim.org has developed a framework to implement large-scale agent-based transport simulations which is called MATSim (Multi-Agent Transport Simulation) [5]. This is used as stand-alone framework. Demand-modeling, agent-based mobility-simulation (traffic flow simulation), re-planning, a controller to iteratively run simulations as well as methods to analyze the output generated by the modules are the things which offers by this framework.

Agricultural value chain management system (iAgri) has been developed by Kumarasiri and Karunananda to cut down the inefficiency and to control the balance in demand and supply chain of foods [6]. According to the iAgri, there are six stages in agricultural value chain such as Deciding, Seeding, Preparing and Planting, Growing, Harvesting and Selling.

A multi-agent system for agricultural stakeholders has also been developed by Adikari and Karunananda [7]. This system contains five agents such as Message Agent, Interface Agent, User Profile Agent, Crop Management Agent, and Selling Agent. This has been developed using JADE environment and can connect though internet.

Information System for Cultivation has been developed by Tharaka and kulawansa [8]. This was developed based on modern information technology infrastructures such as mobile and web services to disseminate domain knowledge to the farmer to enhance the cultivation process. Jayarathna and Hettige have developed a java based distributed communication system (AgriCom) as stand-alone for the Agricultural domain [9]. The system contains four agent namely; Farmer agent, Buyer agent, Seller agent and agricultural instructor agent. That system helps to reduce the communication gap among the agricultural community. The WMAC is the improved web based development of the AgriCom.

### 4. DESIGN

WMAC is a web application which has been developed to reduce the communication gap among the people/stakeholders who are in the agricultural domain. To achieve that objective MAS technology has been used as a concept. The front-end of the system has been developed using HTML and back-end of the system has been developed using PHP and AJAX.

This web application provides four types of users and any person can get registered in any type as he required. The system uses common MYSQL database as ontology. Figure 1 shows the top level design of the web application.

WMAC database is used as ontology of the Multi agent system. This database consists of all the user information and messages. This database also uses as a common message space of the system.

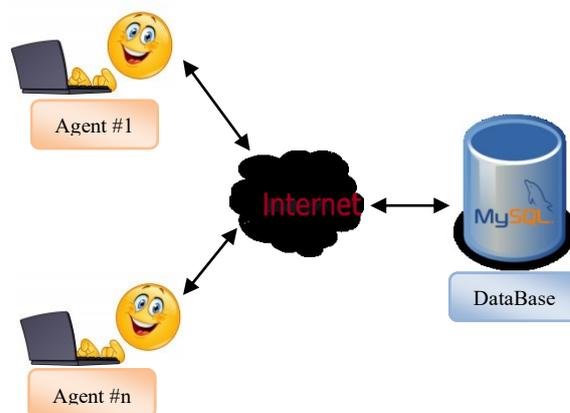


Figure 1: Top level design of the WMAC web application

Figure 2 shows the design architecture of the WMAC agent. WMAC agent consists of three components namely Web Interface, WMAC

Engine and the Web Server. Brief description of each module is given below.

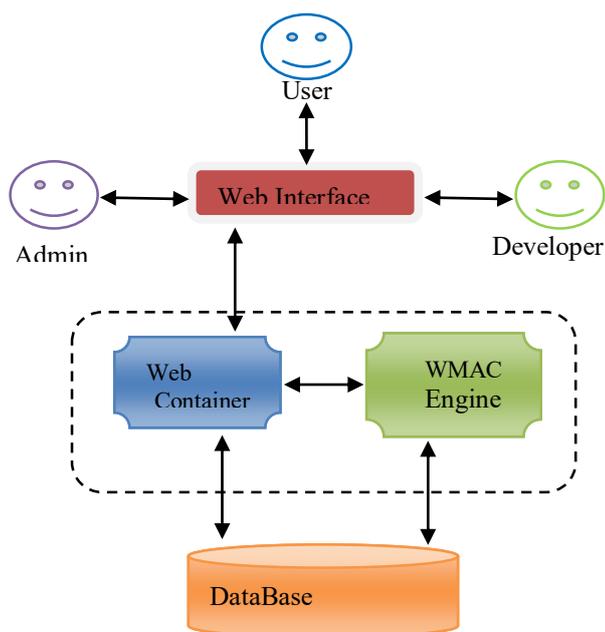


Figure 2: Design of the WMAC

#### A. Web Interface

Agents and the users communicate via web interface. According to the user registered types, features of the web interfaces are different. The figure 3 shows the initial WMAC web interface for all users/ visitors.



Figure 3: Initial WMAC Web interface

After creating a new account user can easily login to the system and can get all the facilities to use the system. At the login, the system automatically set the agent type by using ontology of user.

#### B. WMAC Engine

WMAC Engine is the important part of this

system. This has been developed using the AJAX and is the controlling module of the WMAC which is capable of controlling of the all the features/ tasks that are in the WMAC. WMAC Engine communicates with the Web container and also Database. With the certain time intervals, the WMAC Engine checks the Database and updates the Web container. A user can give particular tasks to do while he logged in and also he can stop that any time. WMAC Engine is the backbone of the WMAC. This makes a communication between web server and the WMAC database.

#### C. Web Container

When the user requests a web page, the browser displays the content of the webpage on the web browser. Web container contains all the web pages. This provides the web pages to your browser. For WMAC, the Apache is used as the web server because of the most valuable features of Apache.

### 5. HOW SYSTEM WORKS

This shows how WMAC works in the practical environment. To use WMAC, users have to run any web browser and go to the WMAC website. WMAC provides all the facilities to members of WMAC. Therefore, at first a new comer has to be a member. For this, a new user has to get registered in the WMAC. Then new user becomes a member as any user type and he can get all the facilities from WMAC according to the system given.

There are two types of working modes WMAC system as manual and automated. If system works in the manual mode then following operation can be done. After registration of new user, he can add new records, send and view new messages and also delete viewed messages. While he is using automated mode, the WMAC system replies to received messages with predefined message structure. Those predefined message structures have been designed at the development stage. In order to provide automated mode, the system time to time check the received messages and read the tags assigned with those messages. Then the system generates new messages and sends them to relevant parties.

In the manual mode each user can set the tasks. After setting the task(s) and change the mode as automate, users can leave from the system. At this point, multi agent system is an active and does the relevant tasks. The figure 4 shows the common message interface for a particular person.



Figure 4: WMAC Common Message Web Interface for Viewers

## 6. EVALUATION

The WMAC has been tested with number of standard testing method with the development. After development of WMAC, this has been evaluated with 22 persons. For this, those persons were allowed to register as their suitable user type with WMAC and to use WMAC for two month period. After two month period questionnaires were distributed among them and collected the results which were showed in table 1.

TABLE 1: Results obtained from questionnaires

Parameter	Very Efficient	Efficient	Satisfactory	Inefficient
Efficiency	3	12	6	1

Parameter	Very Easy	Easy	Satisfactory	Difficult	Very Difficult
User interface understandability	3	10	8	1	0
usability	3	13	6	0	0
Ability to order products	1	3	2	0	0
Ability to sell products	2	2	1	1	0
Ability to buy products	2	2	2	0	0
Ability to sell items	0	3	2		
Ability to instruct	0	3	1	1	0

Parameter	Excellent	Very Good	Good	Poor	Very Poor
Attractiveness	3	13	5	1	

Parameter	Very Useful	Useful	Satisfactory	Not Useful
Usefulness	4	9	7	2

According to the results, most like to use WMAC because this reduces the communication gap among the people who deal with the agriculture.

## 7. CONCLUSION AND FURTHER WORKS

This paper has illustrated the design architecture of the web based multi-agent system, WMAC. This is a web based multi agent system which has an ability to develop communication between persons in the agricultural sector as required. WMAC provides four types of agents as farmer, buyer, seller and instructor agent which represent actual farmers, buyers, sellers and technical instructors in the agricultural community.

WMAC agents work in two modes namely automated and manual. If a user stays with a computer, the agent works with the manual mode and provides facilities to users to communicate each other through the web interface. The system has been tested with 22 users and successful results were obtained. As further work of the project, it is very useful to develop WMAC as a mobile version as this web based system is very much useful for the agricultural domain to reduce the communication gap among the people.

## REFERENCES

- [1] George R., A new direction of research into Artificial Intelligence”, Sri Lanka Association for Artificial Intelligence 5th Annual Sessions, 2008
- [2] Eddile C.L.Chan and Raymond S.T. Lee (2005), iJADE Reporter- An Intelligent Multi-agent Based Context-Aware News Reporting System, Springer-Verlag Berlin Heidelberg. [Online]. Available: [http://link.springer.com/chapter/10.1007%2F11552413\\_99#page-2](http://link.springer.com/chapter/10.1007%2F11552413_99#page-2)
- [3] Catholoin M. Jonker, Remco A. Lam and Jan Treur (1999). A Multi-agent Architecture for an Intelligent Website in Insurance. Springer-Verlag Berlin Heidelberg, [Online]. Available: [http://link.springer.com/chapter/10.1007/3-540-48414-0\\_5#page-2](http://link.springer.com/chapter/10.1007/3-540-48414-0_5#page-2)
- [4] Przemyslaw Kazienko, Multi-agent System for Web Advertising, Wroclaw university of technology, Institute of applied informatics, Wybrzeze S. Wyspianskiego 27, 50-370 Wroclaw, Poland
- [5] MATSim (2013, June 20), [Online]. Available: <http://www.matsim.org/>
- [6] W.C.M. Kumarasiri and A.S. Karunananda, iAgri – An Agricultural Value Chain Management System, Proceedings of the ITRU Research Symposium, Moratuwa, 2010.
- [7] A.A.P.S. Adikari and A.S. Karunananda, Multi Agent Systems for Agricultural stakeholders, Proceedings of the ITRU Research Symposium, Moratuwa, 2009.
- [8] K.J. Tharaka, K.A.D.T. Kulawansa, Information System for Cultivation, Joint National Conference on Information Technology in Agriculture, 2011
- [9] H.M.H.R.Jayarathna (2013), Online Agricultural Communication System (Dissertation submitted to the

Faculty of Information Technology, University of  
Moratuwa, Sri Lanka

- [10] JADE. (2012, June 4). [Online]. Available:  
<http://jade.tilab.com/>
- [11] Jason. (2012, May 27). [Online]. Available:  
<http://jason.sourceforge.net/wp/>
- [12] AgentBuilder. (2012, July 18). [Online]. Available:  
URL:  
<http://www.agentbuilder.com/Documentation/Lite/>
- [13] Zeus Agent System, LOWA STATE UNIVERSITY  
(2013, July 27). [Online]. Available:  
<http://www.cs.iastate.edu/~baojie/acad/current/zeus/zeus.htm>
- [14] NetLogo (2013, July 23). [Online]. Available:  
<http://ccl.northwestern.edu/netlogo/>
- [15] MADKIT (2013, July 29). [Online]. Available:  
<http://www.madkit.org/>
- [16] School of Electronic Engineering and Computer  
Science, Queen Mary University of London (2013, July  
23). [Online]. Available: <http://www.eecs.qmul.ac.uk>
- [17] AgentSheets (2013, May 23). [Online]. Available:  
<http://www.agentsheets.com/>
- [18] Agent Link (2013, August 2). [Online]. Available:  
<http://eprints.agentlink.org/4671/>
- [19] OpenCybele (2013, August 6). [Online]. Available:  
<http://eprints.agentlink.org/5422/>