SRI LANKA ASSOCIATION FOR ARTIFICIAL INTELLIGENCE

Proceedings of the 6th Annual Sessions

30th October 2009

Sri Lanka Association for Artificial Intelligence – 2009

Proceedings of the 6th Annual Sessions
30th October 2009
Moratuwa
Sri Lanka

Editors:

Prof. Asoka S. Karunananda
Dr. (Mrs.) Uditha Ratnayake
Dr. Nalin Wickramarachchi
Dr. Gamini Wijayarathna
Dr. Kapila Ponnamperuma

Sri Lanka Association for Artificial Intelligence, Department of Mathematics & Computer Science, The Open University of Sri Lanka, Nawala, Nugegoda, Sri Lanka

Contents

Keynote Address

Use of Scale Invariant Features in Vision Systems

Dr.N.D.Kodikara
University of Colombo School of Computing

Computer Vision is a field of endeavour attempting to enable man made computers to perceive their environment as humans and animals do. The biological visual process is specifically concerned with making useful models of the environment from a confusing mass of input data sensed by the human eye. Computer Vision attempts to emulate this operation and is concerned with the processing and interpretation of visual data by a computer.

Computers that perceive their environment and perform required tasks could assist in many applications that are routine, tedious, and even dangerous for a human to perform. These applications include industrial assembly and inspection, planetary space exploration, automated medical x-ray screening, monitoring of earth resources by remote sensors, document processing, microscopy and a variety of military applications.

In most computer vision systems, emphasis has been placed on the use of Artificial Intelligence techniques in order to incorporate intelligence into these systems.

Human face recognition

Automated facial recognition is a computer application for automatically identifying or verifying a person from a digital image or a video frame from a video source. This is becoming increasingly important in many areas such as human machine interfaces, multimedia, communication, visually mediated interaction, anthoropomorphic environment and especially in security purposes. Humans have always had the innate ability to recognize and distinguish between faces, yet computers only recently have shown the same ability. In the mid 1960s, scientists began work on using the computer to recognize human faces. Since then, facial recognition software has come a long way. Many face recognition researches have been done regarding still image as well as video based. These approaches include the use of eigen faces, elastic bunch graphs, neural networks and support vector machines. These techniques suffer heavily when the images are scaled, affected by noise and uninformed illumination and distorted due to rotation.

Scale Invariant Features in Face Recognition

This paper presents a new approach to face recognition based on Lowe's Scale Invariant Feature Transform algorithm[1]. SIFT feature, is a robust invariant local feature that has the property of scale and rotation invariance. Also, it is able to provide accurate matching over a certain range of affine transform and illumination changes. Thus, it could be useful to extract distinctive face features over different scales, rotations, views or expressions of human faces for the detection task.

In SIFT approach, four major stages of computation should be followed to generate the set of image features. The first stage of computation is Scale-space extrema detection which searches over all scales and image locations. It is implemented efficiently by using a difference-of-Gaussian function to identify potential interest points that are invariant to scale and orientation. Second stage is Key point localization. At each candidate location, a detailed model is fit to determine location and scale. Key points are selected based on measures of their stability. Third stage is Orientation assignment. One or more orientations are assigned to each key point location based on local image gradient directions. All future operations are performed on image data that has been transformed relative to the assigned orientation, scale, and location for each feature, thereby providing invariance to these transformations. Final stage is Key point descriptor. The local image gradients are measured at the selected scale in the region around each key point. These are transformed into a representation that allows significant levels of local shape distortion and change in illumination. These SIFT features are invariant to scale and rotation and robust to affine distortion, change in 3D view point, noise and change in illumination.

In this research, first SIFT features are extracted for a set of given images. Then they are stored in a separate file and names of the files are stored in a database with details of the relevant person. For a given image, SIFT features are extracted first and compared with the features of the database. Then the best match based on the nearest neighbor approach is found. Figure 1 and Figure 2 show SIFT features and matching key points respectively.

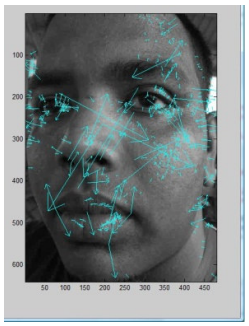


Figure 1: SIFT Features

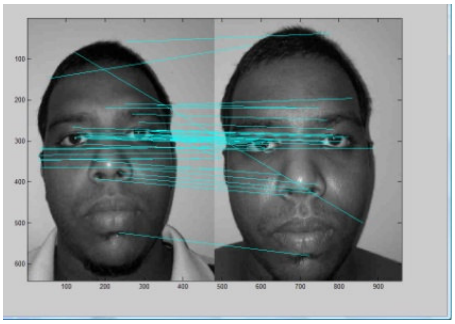


Figure 2: Matching Key points

[1] David G.Lowe, "Distinctive Image Features from Scale-Invariant Keypoints" International Journal of Computer Vision, 60, 2004 pp 91-110

Invited Talk

Game-Based Military Simulations Impact and Current State of the art

Dr. Romesh Ranawana DPhil (Oxon)
Chief Technology Officer, SimCentric Australia
romesh.ranawana@simcentric.com.au

Gaming and simulations based industries have become serious businesses, both economically and socially. In 2008, the net worth of the gaming industry was estimated at 30 billion US$. This explosion in demand and interest has lead to vast amounts of money being spent on the development of better and more realistic simulation and AI engines. This in conjunction with the rapid development of standard, low cost, high-performance PCs has lead to a revival and rejuvenation of the simulation industry. Due to its applicability within its domain and the availability of large amounts of funding, simulations are gaining high popularity and relevance to military simulations. Most modern armies are now composed of dedicated simulation wings, which allocate a large amount of time and effort on the development of realistic and challenging simulations for a variety of purposes.

Military simulations and training have evolved over the years. Initial simulations were based on field exercises and miniature maneuvers on sand tables. In 1958, Charles Roberts and the RAND corporation developed the hexagon-based paper board wargame with combat effects and random number generators [1]. These paper maps were later developed into computerized models to create the first digital battlefield models. Pneumatic flight simulators were first developed during the late 1930's. The 1980s saw the first set of usable computer-driven, 3D immersive, network based combat simulators[1]. The SIMNET[2] system was the first such system to be adopted by the US military for training purposes.

Distributed interactive simulations (DIS) are a revolutionary development which has changed the way that many defense departments around the world operate. DIS is an open standard which allows for the conducting of real-time platform level simulations across multiple platform types. DIS allows simulation developed for multiple platforms and hardware types to communicate with each other and simulate joint operations and missions. For example, the DIS protocol can be used to link a flight simulator controlling a F16 aircraft with another flight simulator controlling a helicopter. The two simulators can be linked together so that they can participate in a joint mission. A collection of soldiers running a networked ground based simulation platform like VBS2 can also participate in the scenario via a DIS link. The possible uses of such collaborative systems are enormous.

Stand-alone or network missions of this nature are used for a whole host of different purposes. Training stands out as the most important use of such systems. Considering both the cost and time saved through the use of simulations in relation to real-world field training, the advantages are insurmountable. A typical training scenario without simulations would involve the practicing of a certain drill multiple times in the field. This would typically involve a large number of resources and time. If a simulation were used in conjunction with on-the-field training, it allows for the same mission to be run as many times as needed within the simulation at a fraction of the cost. When the operatives are deemed to be ready, both physically and mentally, the training can be performed on the field.

Simulations are also used extensively for combat planning and analysis. Different scenarios and operational options can be modeled and evaluated for a fraction of the cost within a simulation environment. Such simulations can be used to provide insights into real-world processes and decision making guidelines. Simulation based techniques are also used as very effectives tools for pre-deployment training. Soldiers are acclimatized with the terrain and environment of previously unseen places before deployment using these techniques.

Simulation based training and testing solutions are also now being increasingly used within numerous other fields. These include emergency planning, medical training, building safety design, city planning etc.

Given the explosion in the use of simulations within numerous fields, the need now exists for the improvement of many of the existing simulation platforms. The primary drawback of most simulation systems is the realism factor. Given a specific situation within a well defined area, it is possible, through extensive programming and design, to simulate a very realistic environment. However, given the vast number of training and analysis types and the different terrain on which they are run, it is becoming increasingly difficult for most simulation companies to accommodate requests. There is therefore an increasingly significant demand for reusable models which can be applied to many simulation environments. For example, there is a current need for a set of reusable urban life generator models. Examples include models for vehicle movement, civilian movement, animal behavior, evacuation and panic behaviors. Such a model would be easily applied to any given terrain by configuring a set of simple parameters. A reusable model of this type could then be applied to any simulation to rapidly construct realistic training scenarios. Artificial intelligence and probability model based solutions are currently the paradigms of choice for such models. Extensive research is being conducted by numerous academic, public and private institutions to develop realistic, reusable and highly configurable models.

References

[1] Roger Smith, "Technology Disruption in the Simulation Industry", Journal of Defense Modeling and Simulation: Applications, Methodology, Technology, Vol. 3, No. 1, 3-10 (2006)

[2] Rheingold, H. (1992). Virtual reality, Simon & Schuster, New York, N.Y.

Mobile Agent for Monitoring Computer Networks

Roshan Kumara[1], Asoka S Karunananda[2]
[1]Center for computer studies, Sabaragamuwa University of Sri Lanka,
[2]Faculty of Information Technology, University of Moratuwa
Email:[1]rolanka@sab.ac.lk, [2]asoka@itfac.mrt.ac.lk

***Abstract* - Network monitoring and maintenance have become crucial when a network comprises of several subnets with heterogeneous hardware and software environments. This paper presents the designing and development of a Mobile Agent based as a solution for monitoring the campus wide local area network in the Sabaragamuwa University of Sri Lanka. The System comprises of three Agents; Host Agent - located in a host, Mobile Agent - movable on the network, and an Agent running on a Mobile phone. The Mobile Agent consists of a stationary unit and a mobile unit. The stationary unit deals with scheduling and controlling while the movable unit travels to locations with faults and does repairing. The mobile agent has also been provided with basic toolkits such as newly updated virus guard, commonly used software, patches, etc. The Agent running on the mobile phone is responsible for sending messages to the administrator to attend to some critical faults. The system has been developed on JADE. JADE-LEAP implements the Agent on the mobile phone with limited resources. The system has been tested on several subnets to investigate the performance and accuracy in real cases.**

Keywords: **Mobile Agent, Computer Networks, Network Monitoring.**

1. Introduction

As networks grow larger, the network becomes more complex. Network monitoring becomes more crucial. In order to maintain and deliver high quality service to end users, network performance and usage must be monitored constantly and potential faults must be dealt with proactively. So the Network Administrators face the challenge of ensuring network efficiency, using of cost-effectively, and providing main services continuously. They need to monitor the whole network system with regard to connection failures and virus infection, if a problem is available, and then manage it while investigating the current status. Computer-based solutions for monitoring networks and communication with the Administrator have been a research challenge.

In recent years, Mobile Agent Technology (MAT) has emerged as a new paradigm for computer network management and monitoring its performance [1]. As a result, a number of research papers involved in MAT in computer network have been published. For instance, Network health care monitoring [2], network configuration management and fault diagnosis [3], Distributed Network Management [4] and Self-Configuring Network Monitor [14] can be cited as few examples. In order to review the technology for monitoring network, the latest trend is to deploy the Mobile Agent to monitor large scale heterogeneous networks. The Mobile Agent is a special object that is autonomous and has the ability to migrate from one node to another node, carrying logic and data, performing actions on behalf of the user. The Mobile Agent concept has many advantages likes autonomy, reactive, proactive, and adaptive capability and collaboration with other agents. After considering all the cases discussed above and current issues of the university network, MAT is more appropriate for monitoring the computer networks and its services. So this paper reports on the design and implementation of Mobile Agent based on network monitoring system (MobiNET).

Accordingly, the MobiNET contains three software agents; the first is Host Agent (HA) and the second which plays significant role in monitoring networks process, is Mobile Agent (MA). The third is in a mobile phone which is connected to the system. The MA has been designed with two main parts. One can live-in on any machine such as server, which is called Scheduling and Control Unit (SCU). It is responsible to manage all the functions of the Network Indicator (NI) and Work Schedule (WS). The other main part is Mobile Unit (MU); it has the ability to migrate to each subnet via LAN for repairing and maintenance of the network. Before its migration, threats from the virus and its possibility are investigated; next the migration process is completed for the maintenance on the host. When the system faces a critical situation, suddenly SCU identifies and subsequently sends the SMS and email which include descriptive information for the cause. The third agent, who lives in the mobile phone, is responsible for monitoring the paoosiblr failures of the MobiNET. If such a failure occurs then the Monitor Agent

sends SMSs to inform the Administrator. In this case, this system has been completely implemented by Java programming language, using JADE tool as an agent development environment.
The rest of the paper is organized as follows. The first section is a brief introduction of the paper. The second section describes related works in monitoring network. The Third section discusses the technology adopted to solve the problem and its appropriateness. The fourth section includes a discussion on the proposed approach to solve the problem. The fifth section reports on designing and implementation of the system. Next section points out the evaluation of the system, while the last section describes conclusion and further works.

2. The State of the Art for Monitoring Networks

An evolution of approaches for monitoring networks has been expanded by the growth in the network speed, bandwidth and technologies. The MA has taken a special place among them. The emergence of MA framework has led many researchers to examine their applicability for network management and monitoring. In this way, it would be interesting to discuss the appropriateness of the MA for monitoring the computer network.

Antonios [16] have developed the methodology for the Autonomous Network Monitoring System (ANMS). This system used traditional method for monitoring network congestion, anomalies, attacks and user define situation. Whenever it happens, the alarm issued by the system can then detect the system in a critical. Distributed Network Management System [4] has been used the MAT and addresses the scalability limitations of centralized network management. Their solution integrates with SNMP for gathering significant information, which is related to the network elements. Huawen Luo [5] has discussed the Agent related network management interaction with SNMP and using JADE as the agent development platform. Further, his discussion focuses on the inefficient problems when managing networks are large in scale, with additional features for programmability and customization of the system. Lee Yip [6] presents system prototype for small scale network management systems by using MAT. It provides the solution via the Internet environment and gives facility to the Administrator to manage and audit the hardware configuration remotely. That concept is more important to manage and update the network configurations remotely without the need of physical contact to the particular computer system. Hong van [7] presents the overall picture of the MA paradigm in computer network; how MAT has been applied in computer network and its applications, especially in fields of network management, resource management, routing and security. In addition to that, Peter Boon [8] has presented theoretical foundations for implementation of the MA system using java. According to the nature of the agent, it clearly indicates that requirement of the MA and its security requirements. Considering the above facts, agent design patterns are pointed out with its implementation using Java as the most suitable programming language.

After considering all these works, the author concluded that still no researcher is interested in producing a monitoring network system which has a dynamic on time communication with the Administrator and managing infections of the computer viruses.

3. Technology Adapted for Monitoring Networks

Agent is a powerful software interaction model that can act autonomously on behalf of its owner and other entities. Mobile Agent is also agent with ability to migrate freely between subnets across the heterogeneous networks. Therefore, MA is a combination of two distinct concepts: mobility and agent. It means, the main idea of mobility in MA is to move self-contained program and data near to a particular subnet. After this migration, MA will execute remotely relevant sets of management tasks, which have been defined according to the nature of the problem and its domain [7, 18]. At the end of these processes, the result will be handed over only to the particular agent. In this sense, this migration and execution results are useful to reduce traffic of the data transmission over the networks [7], that are solutions for the managing and monitoring distributed networks [19].

Features	Description
Mobility	Architecture provides the ability to the agent to move around from its origin location to the destination location to perform the task assigned.
Autonomy	Autonomy refers to the ability to operate without the direct intervention from humans and its own control over their actions and internal state.
Reactive	Reactive refers to the ability to perceive or sense the environment where it is living and to respond in a timely fashion.
Proactive	Proactive refers to the ability to determine the action and decision on behalf of the owners or users.
Adaptive (Learning)	Learning abilities give the agent the ability to adapt and learn from the environment, habits, preferences and working methods of its users.

Collaboration (Social)	Collaboration refers to the ability to collaborate with the other agents and the owners. Upon collaboration, an agent will consider the feasibility execution instruction before proceeding with further action.

Table 3.1: Special Features of the Mobile Agent

The MA has some special features; Table 3.1 describes agent features including mobility, autonomy, reactive, proactive, learning, and collaboration. Certainly, these features are used by the system to support monitoring in a productive manner. According to work in changing environments and react to unexpected events, a MA may need to adapt the current environment to execute the defined process correctly and efficiently.

In all respects, the most important concept of the MA is code migration. The main reason for code migration is to gain performance. For instance, a classic client/server model may consume some network bandwidth caused by all the request/response messages. Moving a process to another machine may reduce this, since only the process is transferred, and possibly, the results afterwards. This model also allows the client on which the agent will run to disconnect from the network after the agent was sent. Not only network bandwidth can be saved, but by sending multiple agents to different machines, parallelism can be achieved in a fairly simple way.

The MA is needed to install a Java Virtual Machine (JVM) as a support agent execution environment at each host the agent needs to visit. That is recommended to avoid moving agents unnecessarily, since this can increase network traffic [8]. Accordingly, there is no argument regarding the suitability of the MA for monitoring heterogeneous network.

4. Proposed Approach for Monitoring Networks

The MobiNET consists of three software agents; the Mobile Agent acts as the major of the system and the second is the stationary agent or the Host Agent which lives-in each connected subnet while the third is a Monitor Agent and it is on the mobile phone which is connected to the system. HA sends a request for the registration when the start-up process of the computer is activated. The MA has the ability to control the system and make it move to remote host for executions. Accordingly, the MA has been designed with two main parts. One can live-in on any machine such as servers, which is called SCU. It is responsible to manage the entire request, manage the NI, and control the WS of the MA. Other main part is MU; it has the ability to migrate to each host via the LAN and is responsible for the repairing and maintenance of the network. But, before each migration, threats from the viruses to the MU and the possibility of its migration are investigated by the SCU. After the migration, the MA can carry out its pre-assigned duties on the subnet. Sometimes MU will be unable to fix a particular problem. If it is a critical situation, then the MA sends an SMS to the Administrator. So it may be very useful to settle the problem immediately with the involvement of an authorized person. System failures of the MN are monitored by the Monitor Agent, and when a failure occurs; immediately the Monitor Agent identifies and sends the SMS message to the Administrator. In the view of proposed system, the MA must be able to carry out several concurrent processes in response to different external events. This implies that the behaviours of the MA must be scheduled cooperatively. The scheduler is implemented based on round-robin non-preemptive policy among all behaviours available in the ready queue.

User- The MobiNET is especially designed and developed for the usage of the Network Administrator by considering all the requirements.

Input- As the major input, data regarding the network status, network services and its failures are captured by the SCU communicating with SNMP agent. Furthermore, SCU sends ping message to each registered host of the MobiNET at fixed time intervals and gets the responses. That will be useful to detect the LAN connectivity and useful to carry on NI.

Process- in our approach, the university network is monitored by the system while maintenance process is continuing. Network monitoring functions investigate any system failures, virus updates and virus infections. The System is looking for any critical situation.

Output- The MobiNET provides the text based monitoring report for each computer and day reports regarding its service status and failures. This would be useful to carry on network maintenance to provide the qualitative and quantitative services to the end users. Additionally, when important subnets, servers, services or part of the network become a critical, the system immediately sends an SMS to the Admin. Concurrently, an email including reasons for the event is also sent to the Admin. Further, the SMS message is sent when system failures of the MobiNET occur. Then, necessary actions can be taken whenever and wherever the Admin is.

5. Design and Implementation of the MobiNET

As shown in Figure 5.1, the top level architecture of the system illustrates each module and its components and interaction with each other for the monitoring of the network process. The system

consists of three main modules: MobiNET (MN), Network Host (NH) and Mobile phone.

5.1 MobiNET (MN)
The MN is the main system and it contains the MA initially. The MA has been implemented with two units; SCU and MU. Always the SCU lives in the MN while MU is migrating for monitoring process.

5.2 Host Agent (HA)
The HA is a stationary agent and it lives-in each host of the network. As stated, the HA activates after the completion of the start-up process of the computer, and sends requests for host registration. In general, each HA is responsible to respond for the ping message from the MA and also sends an acknowledgement after the completion of the required process on the own host. In this sense, HA has limited behaviours to implement the defined tasks.

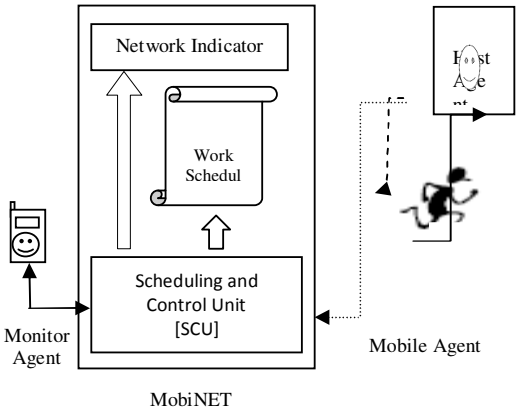


Figure 5.1: Top Level Architecture

5.3 Mobile Agent (MA)
The MA behaves as a system manager and it is the core part of the system, it is used to perform monitoring and controlling operation through its interaction with devices running agent processes. As stated, MA that runs on the MN server contains two modules: the static module is named as SCU and is located in a server, while the other module is MU and has the ability to migrate via network for remote execution, such as scanning for viruses, updating virus guard, removing virus infection, fixing problems.

5.3.1 Scheduling and Controlling Unit (SCU)
As mentioned, scheduling control unit is the stationary part of the MA and that is a new direction for the architecture of Mobile Agent system development. In this domain, after receiving a request message from the HA, it is accepted by the SCU. Then, the particular host will be registered. As a result, SCU updates the WS, & NI, and sends acknowledgment to the HA. Current status of the network is gathered via SNMP that is used to investigate the problems occurred. In the event that several problems occur in different subnets simultaneously, the SCU manages the work schedule concerning the matters such as priorities, its nature etc. Also, the SCU performs various tasks, especially investigation of network connection failures on each subnet and continuation of the NI, maintenance of WS and updating the system log file. After updating of the WS or NI, those events are stored into a database and serialized as the objects that can be used to store as an object when the system terminates due to unexpected errors. When the system starts, it can be de-serialized. In this case, object serialization is useful to store the object settings with java object output and input streams. SCU sends SMSs and an email to the Administrator when the network faces a critical situation. From the perspective of the system, the MA must be able to carry out several concurrent tasks in response to different external events. As stated in our approach, the SCU has been implemented based on round-robin non-preemptive scheduling among all behaviours.

5.3.1.1 Network Indicator (NI)
The NI is one of the components of the SCU, and it indicates an Internet and LAN connectivity of each host, especially web server, mail server, proxy server, DNS server and e-Learning management server. Further, that pays attention to indicate the virus infection and the state of the host. That will be important to continue network maintenance without any mistakes and enhancement of its services.

5.3.1.2 Work Schedule (WS)
The WS of the MA is controlled by the SCU and it is used to schedule the available works which require to be completed. When one is completed by the MA, that will be removed and list of schedule is shifted to upwards by one of them while log file is getting updated regarding the task completed. When urgent message is received, WS will be updated and made its priority also.

5.3.2 Mobile Unit (MU)
The special part of MA is called Movable Unit, and It executes the process required on the remote host according to available faults. If problems exist on to certain subnet, the MU is dispatched to the destination and it fixes the problem immediately. The MA should be protected from the computer viruses when it arrives and leaves the host and we recommend that MU should be scanned for viruses at each stage in the monitoring process. This unit

carries out its assigned duties on the subnet as entrusted by the stationary unit. For this purpose, the MU has been provided with basic toolkits such as newly updated virus guard, commonly used software, patches, etc. The MU performs certain tasks such as reading the WS for available works that need to be completed and then migrates by itself for repairing the connection failures, updating virus guards, scanning for virus infections and removing it without any interventions. After the completion of the required task on the certain subnet, MU reaches to its own server and updates the log file while removing the task which has already been completed, from the WS. The MA will be continuing these processes for monitoring networks. While MU is carrying on the respective jobs as defined, it sends the agent a message including process status to the MA. Then, the SCU understands that it is currently living in and continues defined jobs. Otherwise, as stated in the approach, New MU will be created and sent to the respective host to carry on its task. If MU is unable to reach the particular host, then, MU will be sent to the next host requested by the SCU to carry on the defined processes. After the completion of the task, the log file will be updated and the MU repeats the monitoring process for the next requested host.

5.4 Monitor Agent

The mobile phone has been connected to the system for communicating with the Administrator via sending SMS when the system requires. Failures of the MN are required to monitor for carrying out the monitoring process without any faults independently. Then, Monitor Agent is located on an external phone connected to the system, and responsible for monitoring system functionality. Further, the monitor agent is running in JADE-LEAP environment of the mobile phone. Once a failure in the main system occurs, the situation is identified by the Monitor Agent and an SMS is sent to the Administrator.

The MN has been implemented based on JAVA and JADE as an agent development environment. The eclipse tool is used as a programming code editor with Java (J2SE) and JADE. As stated above, the system has been modeled HA and MA. Agent ID is a unique name of the HA, that can be used to identify the specific agent and its host of the network. Each behaviour of the agent is completed with *jade.core.Behaviour* class. The instance of MA is identified by its code, state information and attributes. The Monitor Agent running on the mobile device was implemented using the LEAP add-on. The LEAF add-on, combined with JADE, provides a modified JADE-LEAF run-time environment for enabling FIPA agent to execute on mobile phone device running Java. It has been developed with consisting of only code excepting data. It especially affects to reduce the storage capacity and enhance the performance with managing available memory of the mobile device [26]. Assuming the communications for network monitoring scenario, MobiNet Ontology has been implemented in compliance with FIPA specification under *mobiNet.ontology* package and contains domain specific vocabulary. Further, defined *MobiNetOntology* gives syntax with different concepts, predicates and agent actions to make a meaningful message structure. SMSLib is a Java integrated API that provides required Java libraries to the system to send SMS text message. MA uses the GSM mobile phone connected via serial port or USB data port to send SMSs. At a critical situation, MA or Monitor Agent uses AT command to instruct the mobile phone to send relevant messages to the Administrator. As stated, the MA sends an email including the reason of the problem.

6. Evaluation

The prototypes developed in this project have been executed for monitoring the network status and identifying critical situations. As stated, a fair evaluation can only be achieved with careful selection of simulation parameters. In this sense, prototype of this system is applied on several subnets of a particular network including servers, in order to check for its reaction with reference to network failures and system accuracy in real life. During the testing period, we have also made some changes in certain subnets and inquire system responses by using system log details and NI.

Host No.	ACL Ping Messages	Disconnected by manually	Detected	Identifying critical	Message Sending
H48H83	192	12	10	8	7
H48H72	188	10	8	1	1
H49H05	192	10	11	2	1
H49H07	189	15	13	2	2
SN49H2	196	10	7	3	1

Table 6.1: Result of an identifying critical situation

Table 7.1 shows results of identifying a critical situation and sending Email and SMS messages. H48H83 is one of the servers and we made network disconnection manually. According to test results, System detects system failures and critical situations successfully and sends the email or SMS. The author defines specific limitations to identify the critical situation of the subnets.

7. Conclusions and Further works

Traditional network management and monitoring based on centralized architecture have been found with feasibility and inefficiency problems. In this sense, we proposed a Mobile Agent based solution. Applying the system, network failures have been evaluated by changing some state of the host and servers manually. Finally, the System records around 80% of system success with the identification of critical situations and sending messages. We discussed the overall system with three agents, one is Mobile Agent and others are Host Agent and Monitor Agent. Ontology of the monitoring network has been defined for agent communication in the system. We have already completed the first two agents and testing of the modules. Development of monitor agent is in progress. The MobiNET has been developed with the use of JADE and eclipse and the system runs on any platform which is connected to the LAN.

We have made some efforts to make the system more successful. Due to this extensive scope of the topic and problem domain, a complete software implementation of the system is beyond the scope of the paper. Implementation of the Monitor Agent and monitoring virus infection on the LAN could be considered as a future work.

7. Reference

[01]. Fabio Bellifemine, Giovanni Caire, Dominic Greenwood, "Developing Multi-Agent Systems with JADE", Telecom Italia, Italy, Whitestein Technologies AG, Switzerland,

[02]. Zapf M., Hermann K., and Geihs K. (1999 May), "Decentralized SNMP Management with Mobile Agents." In Proceedings of IM '99, Boston,

[03]. Ferudin M., Kasteleijn W, and Krause W., (1999) "Distributed Management with Mobile Components." In Proceedings of IM '99, the 6th IFIP/IEEE International Symposium on Integrated Network management, pp857-870, Boston.

[04]. Damianos Gavalas, Dominic Greenwood, Mohammed Ghanbari, Mike O'Mahony, (2000), "Advanced Network Monitoring Applications Based on Mobile/Intelligent Agent Technology", Computer Communications Journal, Vol. 23, No 8,

[05]. Huawen Luo, (2002), "Agent-based Network Management System", M.Sc. thesis, Faculty of Graduate Studies, University Of British Columbia.

[06]. Lee Yip Fei (2003),"Study and implementation of mobile intelligent agent for network monitoring and management", INTI Journal VOL. 1, NO. 3, Pp 193-196.

[07]. Nguyen Hong Van (2004), "Mobile Agent Paradigm in Computer Network", CUC 6th CARnet Users Conference.

[08]. Peter Boon (2005), "Mobile agent systems implemented with Java"

[09]. E.F. de Araújo Lima, P. D. de Lima Machado, J.C.A. de Figueiredo, F.R. Sampaio, (2004), "Implementing Mobile Agent Design Patterns in the JADE framework",

[10]. CERT® Coordination Center, (2002), "A Brief Tour of the Simple Network Management Protocol" ,

[11]. Magid Nikraz1a, Giovanni Caireb, and Parisa A. Bahria (2006), "A Methodology for the Analysis and Design of Multi-Agent Systems using JADE", International Journal of Computer Systems Science and Engineering. pp1-6

[12]. Emmanuel Reuter, Fran¸coise Baude (2002), "A mobile-agent and SNMP based management platform built with the Java ProActive library", http://citeseer.ist.psu.edu/old/reuter02mobileagent.html,

[13]. "Java SNMP Package Introduction" (2008 May), www.gicl.cs.drexel.edu/people/sevy/snmp/snmp_package.html,

[14]. Brian L. Tierney, Deborah Agarwal, Vern Paxson, (2001-2004), "Self-Configuring Network Monitor", A High Performance Network Engineering Proposal: Network Measurement and Analysis For the period June 1, 2001 – May 31, 2004.,(2), pp 3-4.

[15]. Manoj Kumar Kona and Cheng-Zhong Xu, (2007), "A Framework for Network Management using Mobile Agents", IEEE International Conference on Communications 2007**,**

[16]. Antonios Danalis , Constantinos Dovrolis, (2004), "An Autonomous Network Monitoring System", International Journal of Computer Systems Science and Engineering. Pp2-3

[17]. Supranamaya Ranjan , (2006) "Adaptive Mobile Agents: Modeling and a Case Study", International Workshop in Distributed Computing (IWDC 2000), Calcutta,India

[18]. Damir Horvat, Dragana Cvetkovic, (2000), "Mobile Agents and Java Mobile Agents Toolkits", Proceedings of the 33rd Hawaii International Conference on System Sciences

[19]. Shamila Makki, Subbarao V. Wunnava , (2006) , "Application of Mobile Agents in Managing the Traffic in the Network and Improving the Reliability and Quality of Service", IAENG International Journal

of Computer Science, 32:4, IJCS_32_4_16

[20]. Danny B. Lange , Mitsuru Oshima, (2004), "Mobile Agents with Java: The Aglet API", Proceedings of the 33rd Hawaii International Conference on System Sciences

[21]. Joan Ametller, Sergi Robles, and Joan Borrell, (2006), "Agent Migration over FIPA ACL Messages", IEEE International Conference on Communications 2007

[22]. Weiqin Chen and Riichiro Mizoguchi , (1990), "Communication Content Ontology for Learner Model Agent in Multi-agent Architecture", publications from mizoguchi lab http://www.ei.sanken.osaka-u.ac.jp/pub/, Pp 3-7

[23]. Fabio Bellifemine, Giovanni Caire, Dominic Greenwood, (2004), "Developing Multi-Agent Systems with JADE", Telecom Italia, Italy, Whitestein Technologies AG, Switzerland, pp77.

[24]. Asha Nagesh, (2006) "Distributed Network Forensics using JADE Mobile Agent Framework", Student Trade Show and Project Reports, Arizona State University, pp 5-6

[25]. Fabio Bellifemine, Giovanni Caire, Dominic Greenwood, "Developing Multi-Agent Systems with JADE", Telecom Italia, Italy, Whitestein Technologies AG, Switzerland, pp-145-151

[26]. Alf Inge Wang, Carl-Fredrik Sørensen, "A Mobile Agent Architecture for Heterogeneous Devices", Dept. of Computer and Information Science, Norwegian University of Science and Technology, Norway. pp-1-3

Agent Based Solution for Retail Supply Chain Management

Ruvindee Rupasinghe, Sanjaya Ratnayake, Asoka Karunananda
Faculty of Information Technology,
University of Moratuwa
Sri Lanka
ruvindee@yahoo.com

***Abstract*: The entities engaged in the supply chains have become more complex with the expansion of the businesses. Therefore people have used various processes to make the business processes efficient, cost effective and real time. Thus Supply chains have become a main channel of business processes. Supply chain management involves in planning and management of those parties engaged in the chain. Therefore as an intermediary, the retailers' involvement in the supply chains increases the efficiency of the chain. However many problems have been identified in the retail supply chain scenarios such as distribution, distribution strategy, negotiation, communication etc. Due to these problems managing retail supply chains has become a complex problem.**

As a solution, the project implements INSITH, a multi agent system (MAS) which assists in the customer order placing process by choosing the most suitable offer for a particular user. The system uses dynamic intelligent agents; namely, Message Agent, Retailer Agent, Supplier Agent and User Profiler Agent to handle customer requests and provide the best offer by going through various agent negotiation processes. Hence INSITH would be sensitive to the user preferences and will provide solutions based on user experiences and the cost.

1. INTRODUCTION

The retail supply chains consist of suppliers, customers, warehouses, transportation and logistics [7]. The item flow from supplier to customer and information flow from customer to supplier happen through the retailers. Therefore the retailers play a major role in retail supply chains. Further the behavior of a retailer in the business scenario is unpredictable and real time. Therefore in the today's context meeting customer requirements at the given time is a great problem faced by many suppliers. Once the items are ordered the process of delivering the items to the customer involves many complex steps [16]. This process of delivering items to the customer, after sales services, reverse flows such as returning damaged items back to the supplier and handling customer complaints altogether makes the retail supply chain more complex [4]. In each and every step the collaboration of all the components in this chain is highly important to efficiently manage the retail supply process [15, 3].

Nevertheless this process happens in the batch mode. Thus it takes a long time to fulfill the customers' requirements and the supplier fails to deliver items on time. Therefore the supply chain processes have to be highly efficient and need to provide the necessary flexibility to be able to react to short-term changes of the customer demand and unforeseen events during fulfillment [2]. Even though the existing manufacturing software is being used to achieve the expected goals of Supply chains, they have a serious limitation due to the cost of initial investment and resistance to change to new technologies [12]. At the same time they have a serious limitation when it comes to real time processing and forecasting [6]. The reduced negotiation between relevant entities make this process more complicated with the existing supply chain systems. Furthermore they have become inherently complex as each and every component in the chain behaves separately yet try to achieve a common gall [17]. Due to that complexity, the cost incurred in the supply chains has increased a lot. On the other hand the failure in one process has a huge impact on the other processes as they are happening in batch mode [13]. At the same time the customer dissatisfaction could increase a lot due to the supplier failing to meet customer requirements on time with the expected quality. Therefore an agent based approach is suitable for handling this kind of situation [4, 8, 14, 1]. Further it was identified that through proper communication among entities engaged in the retail supply chain the retailers and suppliers can behave more efficiently and meaningfully.

The aim of the proposed solution;INSITH, was to develop a centralized interactive and adaptable system for real time processing of supply chains using the multi-agent technology [9]. INSITH has four agents; namely, Customer agent, Retailer agent, Message agent and Supplier agent who emerge when the request occurs and die when the work finishes. INSITH identifies different scenarios based on the complexity of the transaction and come up with a priority order accordingly.

The rest of the paper is organized as follows. Section two depicts others' work and section three describes the usage of agent technology in effective retail supply chain management (SCM). Section four deals with the design and section five describes implementation approach of the system.

The following section deals with the way INSITH works and section seven describes the implementation approach of INSITH. The conclusion and further work are described in the final section.

2. CURRENT APPROACHES TO EFFICIENT RETAIL SUPPLY CHAIN MANAEMENT

Over the last decade, researchers have introduced various approaches to Retail supply chain management. Cluster approach, hybrid agent based model, trading agents and multi-agent decision support systems are some of them.

2.1. Cluster approach

This research builds an agent-based model to examine retailers' location choice. Here retailers' aim is to maximize profits by changing locations. Two categories of factors impact retailers' decision-making. One is the distance to suppliers, and the other is the distance to markets [5]. This model reveals that clusters emerge, and retailers double up on supplier locations as the number of retailers ascends to a certain number. When more partake in this game, the size of each cluster increases, and cluster density decreases. In this model, retailers directly purchase goods from suppliers and sell them to consumers. It is assumed that players locate in a circular city comprised of discrete locations. The retailers' revenues come from selling products to consumers. Retailers' costs include purchasing cost and shipping cost of products. Retailers' location choice depends not only on their distance to the market, but also on their distance to suppliers. This research thus finds that the centripetal force attracts retailers to supplier locations.

2.2. Hybrid agent-based model to model Petrol market

This is another agent framework based model that uses hybrid agent-based model to a model Petrol retail market [4]. This model was constructed and experiments were conducted to determine whether the trends and patterns of the retail petrol market could be replicated.

Individual petrol stations were created as agent-objects and supplied with knowledge of their initial starting price, production costs, and the prices of those stations within their neighborhood. The prices are either set with real data or idealized data depending on whether the systems are being used to examine real or abstract dynamics. Each agent views the prices of neighboring stations and applies a series of rules to adjust its own prices. Thus the model is shown to reproduce the spatial patterns seen in the real market, as well as well known behaviors of the market.

2.3. Using agent negotiation for effective communication

Another approach can be identified as the use of agent negotiation for effective communication in the supply chain. This framework consists of two types of agents called functional agents and information agents. These agents are usually owned by different companies and are therefore assumed to be self-interested and thus free to join, remain in or leave the supply chain system [18]. Information agents are predefined in the system and help functional agents to find potential negotiation partners or provide other common services such as accepting the registration from a functional agent. All of the negotiating agents have some understanding of system ontology and use a certain Agent Communication Language (ACL) to make conversation.

In the framework, There are no centralized super-agents or distributed mediators to handle the agent cooperation. All these activities occur through negotiation processes, regardless of whether two sides are involved in bargaining for some goods intentionally or de-committing a contract caused by the outside events.

2.4. Implementing trading agents

There are many agent based approaches for SCM by considering it as a game between competing entities for resources. For example the purpose of the KrokodilAgent is to explore how to maximize profit, given the conditions that dominate on the PC market [11]. This has six agents and has its own PC manufacturing companies. The agents are competing in two different markets. On the first market agents compete by buying raw materials necessary to produce personal computers. On the second market the agents are trying to sell all the PCs they produced to customers and at the same time earn as much money as possible. The winner is the agent with the highest bank account at the end of the game. It is also important to establish how the changes during the game effect the game outcome. To play in this game an agent has to connect to the game server. The server has multiple functionalities; it simulates customers and suppliers, controls agent's factory and warehouse and runs the bank.

2.5. Multi-Agent Decision Support System

Another approach for efficient SCM is a decision support agent for SCM scenario [10]. The key idea behind the project is to serve the organizations which deal with a number of activities, such as procurement, production, warehouse management, selling, marketing, and customer services among others, while running a business. To help them to manage these activities, organizations try to automate their business processes. Usually, independent software and hardware solutions are used for each of the activities. However in practice,

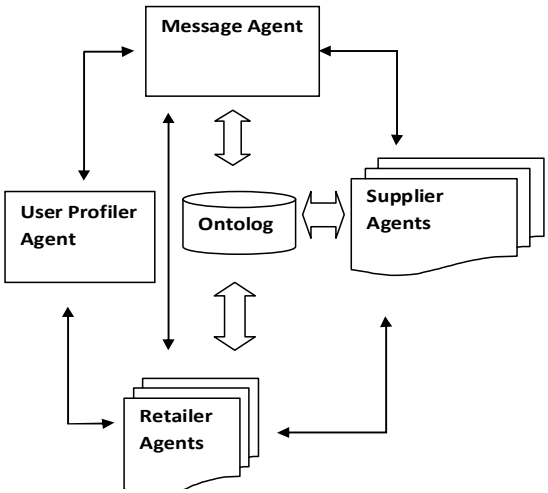
all the activities are highly connected and interdependent.

Since the tasks involved in supply chains are more complex, the aim of the project is mainly focused on the demand part of the supply chain. The expected findings not only can improve a company's performance while running its supply chains, but could also be applied to financial markets and online auctions where the task of predicting winnings and bidding prices is important.

Even though there are various approaches being used, they have failed in efficient modeling of supply chains. Each approach takes the retail supply chain as one scenario and does not take its complexity in to consideration. Therefore it is necessary to identify the various scenarios of the retail supply chains and react to those scenarios in a different manner.

3. APPROACH: MAS FOR TASK BASED MODELING IN SCM

The project implements an agent based retail supply chain to make the processes of the supply chain more effective and efficient. The agent based system makes its processes real time and cost effective. Once the customer places an order, the retailer agents are created. The retailer agents are trying to show their offers and try to compete with the other agents. In this case the most suitable and effective offer will be selected based on the



customers' past experiences and the cost of the offer.

Once the customer places an order, the User Profiler identifies him and retailer agents see it through the message space and check whether they can serve the request. If they have enough stocks and they can serve the request, they immediately pop up their agents and send messages to the message space. All the retailer agents can read the message space. Therefore they can compare their offers with others and check the possibility of their survival in the market. If their offers are high compared to others, they can either reduce the prices or move from that market to another. Other agents can analyze the best profit making business and adjust their strategies to acquire the target market. Each and every transaction happens in the market is recorded in to the ontology in a structured manner so that, later the knowledge gained through the past experiences can be useful. In any case, if the retailers do not have enough stocks with them, they have to go for suppliers. The supplier choosing procedure is as same as the process mentioned above. Here the crucial factors are the time of delivery, past experiences and the cost. There may be some trade-offs depending on the scenario.

Compared to other approaches, there are many advantages of INSITH,such as low cost, reduced time, collaborative decision making on behalf of the user and just in time solutions.

4. DESIGN

INSITH consists of four types of agents namely; User Profiler agent, Retailer agent, Message agent and Supplier agent. Ontology is used as the knowledge base to store communication details of the agents. The high level communication between the entities is described in the Figure 1.

4.1 Agent Implementation

The four agents were implemented in the best way they can negotiate with other agents effectively. As the whole process operates based on the communication between entities, the message agent is playing a key role in this process.

4.1.1 User Profiler Agent

The User Profiler agent is the agent who keeps details of the customers who place orders to the system. There may be many or few customers depending on the situation. Once they place an order, the message is sent to the message space. The user Profiler agents continuously read the message space to see the offers from the retailers. When they see the offers, if they are satisfied with the prices they will buy the items. If they are not satisfied with the offers they will place the order again.

4.1.2 Retailer Agent

Retailer agent keeps listening to the requests coming from customers and creates agents automatically. When customers ask for items, the retailer agents appear with their offers. For example, if the customer requests 5 televisions of the type SONY, there may be many retailers who are having SONY televisions. But their prices may vary according to the services they provide. Therefore, the best retailer may be selected based on their prices and the customers past experiences.

In that case even though many retailers appear, in order to provide the items, only one retailer would be selected by the retailer agent based on the above criteria. In any case if a customer is not satisfied with the current retailer the retailer agents will again work collaboratively to find a different retailer for the customer. Here the retailers can market their items. They can offer special discounts to customers who are buying many items.

4.1.3 Supplier agent

They work as the suppliers or manufacturers who produce and supply items. Whenever a request comes from retailer agents the supplier agents should immediately react to the situation. Producing the items which match the exact requirements of the customers is a major challenge they undergo. Sometimes the suppliers may not have enough stocks. Then they have to negotiate between suppliers to fulfill the order. If the retailer is asking for 200 televisions and Supplier agent is having only 100 items then he has to negotiate with the other suppliers in order to get the rest. So that there will not be any delays of delivery to the retailer. Thus the negotiation between the suppliers makes this process more efficient

4.1.4 Message Agent

Message agent handles the overall communication among entities. Once the customers place an order, the message agent sends it to the message board. Thus through the message board all the agents can view others' status. So Message agents provide good analytical details for retailers and suppliers. Therefore, they can adjust their offers according to market trends and come up with new offers based on the market situation.

4.2 Agent Communication

There are many possible scenarios that could occur in a Retail supply chain. The main entities involved in the process are retailers, customers and suppliers. Once the customer places an order, the retailers would be activated and if there is a matching offer, the Retailer agent would show it to the customer. But this scenario may be complex in some situations. It is possible for many retailers to offer the same offers and then the retailer agent has to identify the best suitable offer for the customer. Further the items that a customer requests may not be with the retailers. Then the retailers have to request the items from the suppliers and further if many suppliers appear at the same time the retailer would have to decide which supplier to select. Therefore, INSITH provides a priority order for each supplier with comparison to the requests from the retailers. The User profiler Agent is able to identify the behaviors of all the retailers and suppliers. Thus, by going through the information from the User Profiler, Retailer agent is able to provide an appropriate supplier order for each retailer request. The communication process of the agents is depicted in figure 2.

When a user logs into INSITH, the user profiler identifies the user and provides the related information which enables the retailer agents to identify the behavior of a particular user. It identifies the user as a retailer or supplier and provides information accordingly. Then the discussion between the retailers and suppliers occurs through the message space. And once it is successful and the order is satisfied and when it is not satisfied they go for alternatives and other options.

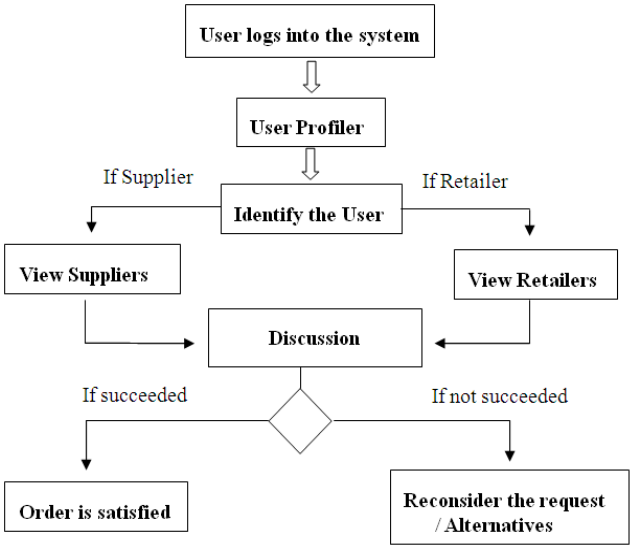


Fig 2: Work flow order of INSITH

Once a customer places an order, the retailer agent will see the order through the message space. Then a retailer would be activated and checks whether he has enough stocks to fulfill the requirement. If many retailers are ready to fulfill the customers' request the retailer agent has to decide which retailer to choose. Here the negotiation among suppliers takes place.

For example, if a retailer places an order asking for 200 computers the supplier agent is activated and checked for the available suppliers. If there is only one supplier for that item and he has enough stocks the retailer can buy from that supplier. In a situation where the supplier does not have enough stocks, the retailer has to look for alternatives. Further complexity increases when the retailer is requesting many types of items at the same time. If the retailer is requesting 200 computers from the supplier and if one supplier fails to deliver the whole amount two suppliers can collaboratively satisfy the offer. Thus when the retailer is requesting many types of items many suppliers can collaborate to fulfill his requirement

4.3 Managing the priority order

The priority order is maintained by providing a relative weight for each supplier for each retailer's requests. INSITH allows users to specify the item information such as lead time, amount, number of items, expected quality and type of item etc. Further, it keeps the relationship between each retailer and supplier. The history information is maintained by tracking how many times a particular supplier has rejected a particular user or item and how many times they have done transactions.

Thus INSITH allocates a relative weight for each parameter and the total weight is measured using the total of all the relative weights. Further, INSITH considers the complexity of the scenario. The selection criterion differs according to the number of retailers and suppliers available.

5. IMPLEMENTATION

The agents have been developed using Java Agent Development Environment (JADE). JADE is implemented according to the FIPA specification and therefore as jade has been used as the programming platform the agents are also implemented according to the FIPA specification. FIPA specifies three major concerns for developing agents. Those are agent communication, agent management and agent message transport. Here each agent has been implemented as a separate java class. Their behaviors are defined within the class scope. The communication methods are implemented in the message agent.

5.1. Agent Negotiation

The whole process happens through the agent negotiation. The agent negotiation is implemented in a way similar to the real world. With relation to real world retail supply chains there are many retailers who are engaged. They have their own prices and they are willing to negotiate and reduce prices and change their offers according to requests. Some retailers are giving discounts when they feel they are having excess stocks or demand decreases. Retailers are interested in regular customers and try their best to keep them satisfied. Sometimes it may be profitable for retailers to interact with the customers who are buying large stocks and in such scenarios they may ignore small requests. Further if others offers are low cost and if one cannot exist in the market they will disappear or change their prices to match the current market conditions. Therefore at the retailer agent negotiation stage, each retailer agent is given a weight according to his preferences to indicate its strength to match a particular request. Parameters like type of request, expected quality, distance to market and user buying history, number of items, item type and cost of items are considered in defining the weight. The weight is defined with a request. Therefore same agent may have different weightings for different requests. The most appropriate agent for each request is given the highest priority while others are given less weights. So the agents who are given the highest weight value will prompt to serve the request. Thus the wight allocation is done according to the best supplier measuring algorithm.

5.2. Best supplier measuring

Quantifiable parameters such as number of items, item type, lead time, cost of items, distance to market and non-quantifiable parameters such as type of request, quality, and user buying history are considered in defining a weight. The non quantifiable parameters are measured with an index value. The weight of a supplier is created with a request. All suppliers are given a specific value for each parameter with comparison to the retailers request and the total of parameter value is considered as the total weight of that particular supplier. Therefore same agent may have different weightings for different requests. The best supplier measuring algorithm is depicted in figure 3. The most appropriate agent for each request is given the priority value 1 while others are given 0. So the agents who are given priority value 1 will be given first priority and prompt to serve the request. The priority allocation for the requests is done using the following algorithm (figure 3).

```
int totalMark, supplierQuality, retailerQuality,
supplierPrice, retailerPrice, supplierLeadTime,
retailerLeadTime, supplierAmount,
retailerAmount,numberOfTimesRetailerRejectSupplier,nu
mberOfSuppliers,
numberOfTimesRetailerBuyFromSupplier;
String bestSupplier;
if (retailerLeadTime == supplierLeadTime) then
        totalTotalMark += 100;
else if ((supplierLeadTime - retailerLeadTime)==1) then
        totalMark += 75;
else if ((supplierLeadTime - retailerLeadTime)==2)  then
        totalMark += 25;
else if ((retailerLeadTime - supplierLeadTime)==1) then
        totalMark += 80;
else if ((retailerLeadTime - supplierLeadTime)==2) then
        totalMark += 50;
if (supplierAmount == 0) then   totalMark -= 200;
else if ((supplierAmount - retailerAmount) < 200) then
        totalMark += 100;
else if (((supplierAmount -retailerAmount)<1000)&&
      (supplierAmount - retailerAmount)  > 200) then
totalMark += 75;
else if ((supplierAmount - retailerAmount)>1000) then
totalMark += 25;
if ((retailerPrice - supplierPrice) < 500) then   totalMark +=
100;
else if (((retailerPrice - supplierPrice) > 500) && (RetP-
SupP) < 1000) then  totalMark +=25;
else if ((retailerPrice - supplierPrice) > 1000) then
totalMark +=75;
else if ((supplierPrice - retailerPrice) > 1000) then
totalMark +=0;
else totalMark += 10;
```

```
for numberOfTimesRetailerBuyFromSupplier totalMark += 20;
for numberOfTimesRetailerRejectSupplier totalMark -+= 20;
for numberOfSuppliers bestSupplier = max (numberOfSuppliers)
```

Fig 3: best supplier measuring algorithm

In the algorithm for each scenario, each retailer is given a weight. If the suppliers cost of item exceeds the retailers expected value of the same item, in such scenarios it is unlikely that they will do a transaction. Further if they haven't done transactions before, INSITH provides a low priority for the particular supplier to satisfy that retailer. But the same supplier may give a higher priority when doing transactions with another retailer. The algorithm provides a low priority for the new comers with comparison to the experienced suppliers and retailers. Therefore, with the time, the new comers may also lead in the market. More importantly if a supplier is providing high cost items with low quality and with a delay of supply, no retailers will go for that particular supplier. Then they will be given a low priority and they will vanish from the negotiation process while more reasonable and effective agents will remain there.

6. HOW INSITH WORKS

When a user logs in, the system identifies the user as a retailer, supplier or a customer through their log in information. Then he will be given the available offers, offered by various retailers and suppliers. This information is filtered by the message agent based on the users' preferences and recent activities. For example Retailer A wants to buy 10 SONY Televisions with the cost of Rs. 12,000. So the Retailer A is prompted with the available Suppliers who are offering SONY televisions with the price close to the retailer's request. The Supplier agents are waiting for the requests from the retailers and try to sell goods to them. Here the suppliers are competing with retailers and the retailers are also competing for suppliers. Thus the best solution emerges as an emergent property of the agent negotiation process.

Agent negotiation results depend on the parameters it passes and the content it has absorbed previously. For example, Retailer agents will look for sellers considering factors such as price, quantity, quality level, past transactions and relationships (INSITH will give a weight based on the customer satisfaction), lead time (time taken to deliver goods from the date of ordering), the extent to which the order quantity matches with the Economic Order Quantity, and taxation transportation and other expenses. More importantly, customers can prioritize the above mentioned requirements. Then Retailer agent will look for the sellers who match with the mentioned parameter. For example, one customer may want to purchase the cheapest car irrespective of the quality. In order to execute this request Retail agent will move through all the archives to find the lowest offer. Mean time if the customer requires a quality product, then Retailer agent will take quality as a parameter too.

7. CONCLUSION

At the moment the project has been concentrated on retail market where few customers and retailers are available. But later this study can be expanded to handle huge SCM environments having thousands of customers, retailers, wholesalers and suppliers. Furthermore, features like warehouse management, procurement management, logistics management also can be added.

When the businesses grow the business transactions occur in them would be much complex. There would be thousands of records stored in the ontology. Data processing would be much more complex and as a result knowledge base will expand heavily. But with the use of agent technology users will be able to work in such environments efficiently. As discussed throughout the report, the adapted approach is more users specific and based on agent technology. When compared with other systems this has a unique feature being able to adjusted itself to different scenarios. Further the system can be updated with new diverse agents.

ACKNOWLEDGEMENTS

All the faculty members and our colleagues deserve to be appreciated for contributing to the success of this project in various ways. Also the authors of the references we have used throughout this project are highly appreciated.

REFERENCES

[1] Forget, P., Monteiro, T., D'Amours, S., Frayret, J. M., "collaborative agent based negotiation in supply chain planning using multi-behavior agents"

[2] Frey D., Stockheim T.,Woelk P. O., Zimmermann R., "Integrated Multi-agent based Supply Chain Management", 1st International Workshop on Agent-based Computing for Enterprise Collaboration , Linz, Austria 2003

[3] Fu, Y., Piplani, R., "Agent-based simulation applications: multi-agent enabled modeling and simulation towards collaborative inventory management in supply chains", Proceedings of the 32nd conference on Winter simulation Orlando, Florida , 2000,Pp 1763 - 1771

[4] Heppenstall, A., Evens, A. Birkin, M.,"Using Hybrid Agent-Based Systems to Model Spatially-Influenced Retail Markets", Journal of Artificial Societies and Social Simulation vol. 9, no. 3, 30-Jun-2006

[5]Huang, A., Levinson,D., "An agent-based retail location model on a supply chain network", 2008

[6] Johnson M.E., Pyke D.F., "Supply chain Management Integration and Globalization in the Age of eBusiness", Tuck School of Business at Dartmouth, 2001

[7] Johnson M.E, Pyke,D. F., "Supply Chain Management", The Tuck School of Business, Dartmouth College Hanover, July 27, 1999

[8] Jennings, N.R., Wooldridge, M., "Applying Agent Technology, Journal of Applied Artificial Intelligence special issue on Intelligent Agents and Multi-Agent Systems", 1995

[9] Kok,J.R., Vlassis,N., "Collaborative Multiagent Reinforcement Learning by Payoff Propagation", Journal of Machine Learning Research, vol 7 Jan 2006

[10] Kovalchuk, Y., "Multi-Agent Decision Support System for Supply Chain Management", The Supply Chain Management. Game for the 2007 Trading Agent Competition,Technical Report CMU-

[11] Petrić,A., Jurasović, K. "KrokodilAgent: A Supply Chain Management Agent" , 8th International Conference on Telecommunications – ConTEL 2005, June 15-17, 2005

[12] Potok T.E., Ivezic N.D., Samatova N.F., "Agent based architecture for flexible cell design, analysis and evaluation", 4th International Conference on Design of Information Infrastructure Systems for Manufacturing, 2000

[13] Santa-Eulalia, L.A., D'Amours, S., Frayret, J.M., "A methodological framework for the analysis of agent-based supply chain planning simulations", Proceedings of the 2008

[14] Sanya, L., Hongwei, W., "Agent Architecture for Agent-based Supply Chain Integration & Coordination", ACM SIGSOFT Software Engineering Notes Volume 28, Issue 4 ,July 2003

[15] "Supply chain transformation is becoming a critical element for driving business results", Insights from the 2006 Supply Chain Executive Summit, IBM and Stanford University

[16] Verdicchio, M., Colombetti, M., "Commitments for Agent-Based Supply Chain Management", ACM SIGecom Exchanges, Vol. 3, No. 1, 2002.

[17] Vickers P., Kodarin A., "Deriving Benefit From Supply Chain Complexity", PRTM, 2006

[18] Ye Chen, Yun Peng, Tim Finin, Yannis Labrou, and Scott Cost, "Negotiating Agents for Supply Chain Management", Computer Science and Electrical Engineering, University of Maryland Baltimore County, August 10 1999.

Ontology Based XML Schema Generation

Kanishka Jayawardene kanishka_kan@yahoo.com
Kapila Ponnamperuma kapila@kln.ac.lk
Shantha Jayalal shantha@kln.ac.lk

Department of Industrial Management
University of Kelaniya, Sri Lanka

Abstract :There are many commercial and non commercial tools available for XML Schema generation for XML instance documents. However most of the tools produce the schema by guessing the metadata from the instance documents. Therefore lots of manual works are needed to fit the generated schema to match with the real requirements. This paper presents a different approach to generate XML schema using domain ontology with very high accuracy and completeness.

Keywords: XML, Ontology, XML Schema.

1. Introduction

EXtendible Markup Language (XML) is a simple and flexible language to represent data and was found in 1996 [3]. XML inherits the qualities of both HTML and SGML (Standard Generalized Markup Language). World Wide Web Consortium (W3C) ratified the XML in 1998 [7]. Since then XML has been evolved and been used in many areas such as data storage, messaging, interfacing, etc. Even though XML is a simple language, now it has a huge impact on the growth of the Web. XML is the basis for several next generation web technologies. Such as XHTML, RSS (Blogs), AJAX, Web Services [6].

This has lead to the creation of a huge amount of both on and off line XML documents for many purposes such as exchanging data or messages between different applications. These applications also require corresponding XML schema definitions with each of these XML instance documents for the purpose of validation [4]. Generating these XML schemas for each XML instance is a laborious task which needs a lot of human expert time [6].

2. Related Works

There are a number of tools available to create XML Schemas from XML documents. Tools such as Visual Studio XMLSpy [12], Stylus Studio [11] etc. are capable of generating XML Schemas automatically. Though different tools are able to generate XML schemas automatically, still those tools lack the capability of producing schema with high accuracy and with matching requirements [4]. It will be correct for the document but there might be instances where the document can vary while still remaining valid. And almost all available tools often generate a best-guessed schema from any XML document.

These tools make certain assumptions about the structure of the xml document, based on the data found in the XML document. For example, it will always set `minOccurs` to `1` and `maxOccurs` to `unbounded` for each element. It will also always use the `xs:sequence` compositor for lists of elements, even if the XML document has elements in various orders. For these reasons, it is not possible to take the schema generation granted. It has to be edited always to make sure it will fit in to the real requirements.

Method used in [4] uses a domain model by means of database tables. However, the main problem with the database approach is that though it works fine for a single enterprise, database structure may have to be changed for a different enterprise.

3. Objectives of the Research

The purpose of this research is to develop an automatic schema generator for XML documents using domain ontology. The ontology is used in order to provide the domain knowledge such as terminology, restrictions and data types. The XML schema generator can be used in any domains provided that the ontology is available for that domain. This will have a great impact on applications such as communication systems where XML is widely used. The manual work will be reduced in business organizations. Once the ontology is created for an organization, it is possible to refer the terms used in the ontology in XML instance documents and get the schema generated automatically.

4. Methodology

Most existing tools are focused on predicting the accurate XML schema depending on the data within the XML file. Though the XML schemas can be predicted, it is not the most accurate schema. The reason for this is mainly due to the absence of the domain knowledge in which these XML documents are created.

We focus our research to overcome these problems by proposing a general ontology [9] as the domain model. Given the ontology, XML schema can be generated for any XML document created within the assumed domain model.

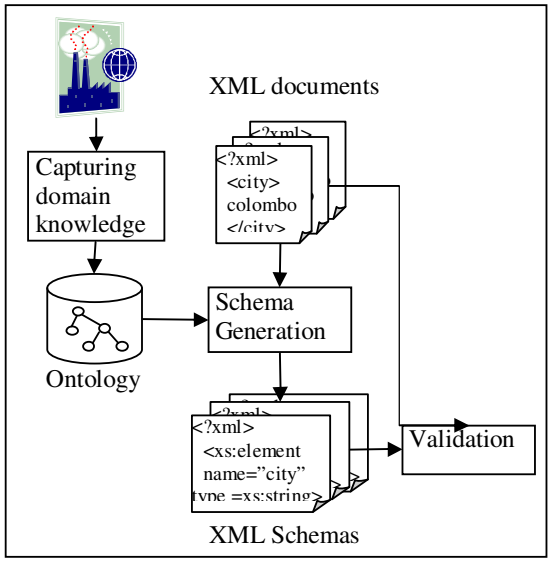


Figure 1: The conceptual model

Capturing Domain Knowledge

In order to create XML schema automatically for every XML document, it is necessary to store the domain knowledge related to the specified XML file in some machine-readable form. Ontology was selected as the means of capturing and storing the domain specific knowledge due to the flexible nature and the reasoning ability. Particularly RDF was chosen to develop the ontology as this had been used successfully in many web based applications. As this research mainly focuses on capturing terminology and related semantics used in a particular domain, methodology used in [8] has been used to define the concepts and their relationships. As the first step, the ontology should be developed, as it contains the terms used in the particular domain and its types and relationships.

Schema Generation

After defining a framework for the abstraction of the ontology, the next step was the algorithm development. The objective of the algorithm was to process any valid XML document and to produce the XML schema by obtaining necessary knowledge from the given ontology. *Appendix A* shows the abstract algorithm used in the implementation. If the XML document does not comply with the ontology, XML schema will not be produced as we use the closed world approach for ontology reasoning.

5. Implementation and Results

Protégé [2] ontology development tool was selected to develop the ontologies used in our prototype development due to its wide acceptance and the strong community support. Figure 2 shows the class hierarchy for the sample University ontology created using Protégé.

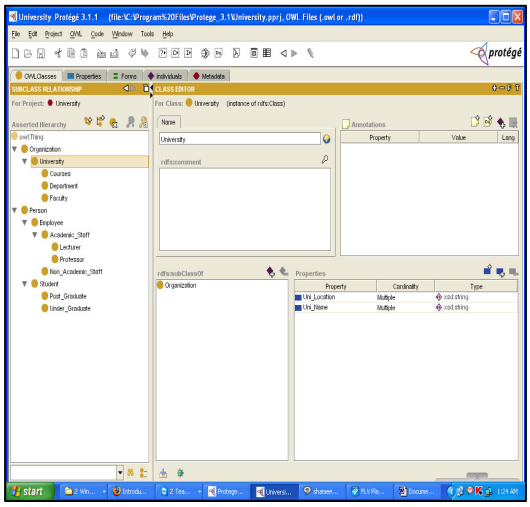


Figure 2: Class hierarchy of the university Ontology

XML Schema Generator

XML Schema Generator tool has been implemented using C#.Net. A screen shot of the tool is given in figure 3. In order to read the RDF triples written by Protégé, it was necessary to use a third party library since C#.Net does not have a precompiled library. Therefore, SemWeb [1] which is a .NET library designed for working with RDF is used. It provides methods for reading, writing, manipulating, and querying RDF.

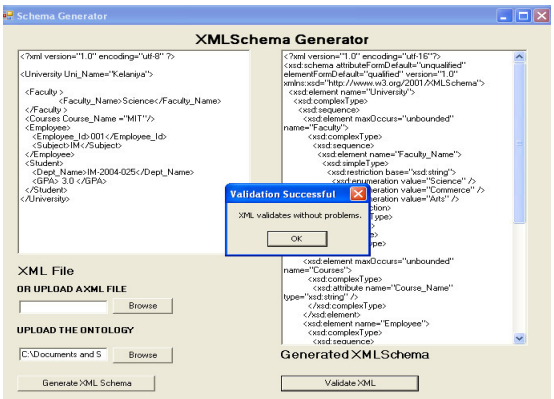


Figure 3: Screen shot of XML Schema generator

XML Schema generator is capable of generating the XML schema for any given XML document. The Ontology should also be provided in order to acquire the domain knowledge. For the simplicity in this research paper, a sample ontology of a University is mentioned. Appendices *B* and *C* respectively show a sample XML document and the corresponding schema generated using the XML Schema Generator tool.

6. Evaluation

Evaluation of our model has been done in two ways. First the generated XML schema and the XML instance documents were validated using standard XML schema validation schemes. Initially, a syntax check is performed while uploading the document. If the XML file is not syntactically correct, an error is given saying the XML is not well formed. After generating the Schema for a well formed XML file, the XML document is validated against the generated schema. Secondly, human experts also used to check whether the domain requirements such as data types, cardinality constraints, etc. have been captured correctly into the XML schema. During the validation, several ontologies (sample shown in *Appendix C*) have been used and the result has been checked with the existing tools in order to assess the accuracy of the generated schema.

In order to evaluate the accuracy of our model we used several XML instance documents (sample shown in *Appendix B*). Each XML document consists of five elements and four different XML documents were considered from each ontology for the evaluation. Generated data types and the number of occurrences from each tool is compared with the expert. *Appendix D* shows the comparison between the XML Schema generation tools.

The comparison shows that there is a 30% improvement on average for the data types and 48% improvement on average for the number of occurrences feature. Therefore, there is a significant improvement in accuracy.

The existing tools are not capable of generating restrictions. But, our model is capable of generating accurate restrictions and hence shows a 100% improvement in completeness when compared to the existing tools available.

7. Conclusion

The main advantage of generating XML schema automatically is that this method can be used in applications where many XML files are generated daily and compliance schema is required. Evaluation shows that our tool can be used to generate XML schemas accurately for a domain specific XML documents. The increased accuracy of XML schemas probably makes the communication more effective. It is possible to use this in many industries without being restricted by the distance and platform.

8. References

[1] Semantic Web/RDF Library for C#/.NET By Joshua Tauberer
http://razor.occams.info/code/semweb/

[2] Getting Started with Protégé-OWL (3.x series)
http://protege.stanford.edu/

[3] W3C XML Schema Definition Language
http://www.w3.org/XML/Schema

[4] Lisa Bahler, Francesco Caruso, Josephine Micallef, *Experience with a Model-Driven Approach for Enterprise-Wide Interface Specification and XML Schema Generation*, Proceedings of the 7th International Conference on Enterprise Distributed Object Computing, September 16-19, 2003

[5] W3C Namespaces in XML
http://www.w3.org/TR/1999/REC-xml-names-19990114/

[6] W3C Extensible Markup Language (XML)
http://www.w3.org/XML/

[7] Bray T, Sperberg-McQueen CM, Paoli J, & Yergeau F. *Extensible Markup Language (XML) 1.0.* http://www.w3.org/TR/REC-xml/ . 2007.

[8] Deen SM & Ponnamperuma K. *Dynamic Ontology Integration in a Multi-agent Environment.* 20th International Conference on Advanced Information Networking and Applications (AINA 2006)., 367-372. 2006. IEEE Computer Society.

[9] Noy NF & McGuinness DL 2001, *Ontology Development 101: A Guide to Creating Your First Ontology* Stanford Knowledge Systems Laboratory Technical Report KSL-01-05.

[10] POL. Protege Ontology Library. http://protege.cim3.net/file/pub/ontologies/travel/travel.owl . 1-5-2007.

[11] Stylus Studio 2009
http://www.stylusstudio.com/

[12] XMLSpy 2009
http://www.altova.com/xml-editor/

Appendix A

```
Load the Required Ontology to the memory
Extract the Root element from the XML file
  CreateComplexType (XmlElement)      {
            Create an Array List
            Check the Element for Child nodes
             For each Child Node {
                        If the node is an Element                    {
             If no existing Node as the same name in the Array List
                        {
                        Add the new node to the Array List
                        Sort the Array
                        //Node contains Children or Attributes
Create a Complex type container
                        If (Node has ChildNodes OR Attributes)      {
                                    CreateComplexType (Node)
                                    }
                                    Else{
                                                //No complex Type
                                    Needed
                                    Create a XMLSchema Element
with the Node name
                                    }
                        Check the Ontology for the maxOcuurance
                        Set the Xml Schema Element
maxOccurance
                        }
                        }
            }
            If the element has Attributes {
                        For each Attribute {
                                    Create XML Schema Attribute
                        }
            }
            If the Element is Complex Type {
                        Add Complex Type Name
            }
                        Else {
                        Add a Simple Type
                        Check the Ontology for Simple Type
Restriction
                        Check the Ontology for Data Type
            }
                Return XML Schema Element
        }
 Create the XML Schema
```

Appendix B

```
<?xml version="1.0" encoding="utf-8" ?>
<University Uni_Name="Kelaniya">
  <Faculty >
<Faculty_Name>Science</Faculty_Name>
  </Faculty >
  <Courses Course_Name ="MIT"/>
  <Student>
    < Student_Name>Kanishka</Student_Name>
    <GPA> 3.0 </GPA>
  </Student>
</University>
```

Appendix C

```
<?xml version="1.0" encoding="utf-16"?>
<xsd:schema              attributeFormDefault="unqualified"
elementFormDefault="qualified"                version="1.0"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="University">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element maxOccurs="unbounded"
name="Faculty">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="Faculty_Name">
                <xsd:simpleType>
                  <xsd:restriction base="xsd:string">
                    <xsd:enumeration value="Science" />
                    <xsd:enumeration value="Commerce" />
                    <xsd:enumeration value="Arts" />
                  </xsd:restriction>
                </xsd:simpleType>
              </xsd:element>
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
        <xsd:element maxOccurs="unbounded"
name="Courses">
          <xsd:complexType>
            <xsd:attribute name="Course_Name"
type="xsd:string" />
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="Student">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="Student_Name"
type="xsd:string" />
              <xsd:element minOccurs="0" name="GPA"
type="xsd:float" />
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
      </xsd:sequence>
      <xsd:attribute name="Uni_Name" type="xsd:string" />
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

Appendix D

		No of occurrences				Improvement
Ontology		**Visual Studio**	**XML Spy**	**Schema Gen**	**Expert**	
University	Test case1	3	3	5	**5**	
	Test case2	3	3	5	**5**	
	Test case3	3	3	5	**5**	
	Test case4	2	2	5	**5**	
Match with the Expected		**11**	**11**	**25**		**56%**
Vehicle	Test case5	3	3	5	**5**	
	Test case6	4	4	5	**5**	
	Test case7	4	4	5	**5**	
	Test case8	4	4	5	**5**	
Match with the Expected		**15**	**15**	**25**		**40%**

		Data type				Improvement
Ontology		**Visual Studio**	**XML Spy**	**SchemaGen**	**Expert**	
University	Test case1	5	5	5	**5**	
	Test case2	4	4	5	**5**	
	Test case3	3	3	5	**5**	
	Test case4	3	3	5	**5**	
Match with the Expected		**20**	**20**	**25**		**20%**
Vehicle	Test case5	3	3	5	**5**	
	Test case6	3	3	5	**5**	
	Test case7	3	3	5	**5**	
	Test case8	3	3	5	**5**	
Match with the Expected		**15**	**15**	**25**		**40%**

On Demand Web Page Translation -BEES in action-

B. Hettige[1], A. S. Karunananda[2]
[1]Department of Statistics and Computer Science, Faculty of Applied Science, University of Sri Jayewardenepura, Sri Lanka.
[2]Faculty of Information Technology, University of Moratuwa, Sri Lanka.
budditha@yahoo.com[1] , asoka@itfac.mrt.ac.lk[2]

Abstract **–Web-enabled technologies including www, email are widely use and have become popular communication media in the modern world. However, many of these services are available only through the English language. This is a problem faced by millions of internet users who are not fluent in English. Therefore, many countries address this issue by using Machine Translation technologies to translate these English based web resources into their local languages. This paper reports the design and implementation of the English to Sinhala Machine Translation system (BEES) that has been developed to translate an English web page in to Sinhala through the concept of Varanagema (conjugation) in Sinhala Language. In addition, it uses a context-based approach to semantic handling. The design, implementation and major translation issues have been presented in the paper.**

Introduction

World Wide Web (www) is the most widely used and popular communication media in the modern world. From a technical viewpoint, it is a system of interlinked hypertext documents accessed via the Internet [19]. There are so many services and facilities available on the internet such as web, email, chat, forums, Facebook etc. It should be noted that, many of these services are available only in the English language. This is a problem for millions of internet users who are not fluent in English. The obvious solution for this issue is the use of modern computing technologies to translate English to local languages. This is call machine translation (MT). The machine translation is a sub field of Natural Language Processing (NLP), which is one of the most achieved areas in Artificial Intelligence (AI).

In Sri Lanka, Sinhala language is spoken by about 16 million people. Sinhala is one of the constitutionally-recognized official languages in Sri Lanka, along with Tamil. However, 80 % of Sinhala speaking people do not have the ability to read and write in English well. Therefore, the development of a English to Sinhala Machine translation system is a highly valuable product for all Sinhala speakers who are not fluent in English language.

Nowadays, thousands of Machine translation systems have been developed for different languages. Among others, Apertium [30], Google Translate [28], Babel Fish [25] and SYSTRAN [22] are well-known machine translation systems in the world. In the region, Anusaaraka [2], AnhalaHindi [4], ManTra [6], AngalaBaratha [5], English to Urdu machine translation system [35] belong to the Indo-Aryan family [19] of machine translation systems. On the other hand, perhaps, EDR [36], the machine translation system by the Japanese is the most completed system so far.

These translation systems use various approaches to machine translation, including, Human-Assisted translation, Rule based translation, Statistical translation and Example-based translation. However, due to various reasons associated with the complexity of languages, Machine Translation has been identified as one of the least achieved areas in computing over the last sixty years. Most of these issues are associated with semantic handling in the machine translation systems.

We have been working on a project to develop an English to Sinhala Machine translation system namely BEES. The BEES is acronym for Bilingual Expert for English to Sinhala. It has been powered by theory of Varanagema (conjugation) in Sinhala language. In this project we have already developed a Sinhala parser [7], intermediate-editor [11], Sinhala morphological analyzer [8], three lexical dictionaries [9] and Transliteration module [10]. Each of these modules and their prototype integrations have been tested through several real world applications namely Human-Assisted machine translation system [11], web-based selected text translation system [12] [13] and context-based machine translation system [14].

This paper reports a new version of the BEES that can translate a given web page into Sinhala. This system works based on the concepts of Varanagema and handles the semantics of the sentence through the context-based approach.

The rest of this paper is organized as follows. Section 2 gives an overview of some existing machine translation systems. Section 3 reports a brief description about previous developments of the BEES. Then section 4 gives the design of the

translation system. Section 5 discusses current issues in the English to Sinhala Machine translation and section 6 shows how the system works for the given web page. Finally, Section 7 concludes the paper with the conclusion and a note on further work.

Brief review of the Mchine Translation

The Machine translation approaches can be classified into three categories, namely, statistical approach, example based approach and rule-based approach [19]. The Statistical approach uses some statistics such as mean, variance on bilingual text corpora to find the most appropriate translation. The Example-based approach is often characterized by its use of a bilingual corpus with parallel texts as its main knowledge base. The rule based approach requires extensive lexicons with morphological, syntactic, and semantic information, and large sets of rules. Therefore, any rule-based machine translation system contains a source language morphological analyzer, a source language parser, translator, target language morphological analyzer, target language parser and several lexicon dictionaries. Further, in relation to English to Sinhala machine translation, the system needs an English dictionary, an English-Sinhala bilingual dictionary and a Sinhala dictionary.

A large number of machine translation systems have been developed under the above three broader headings.

For instance, Apertium [30] is a rule-based MT system that translates related languages. This is an open –source system that can be used to translate any related two languages. This MT engine follows a shallow transfer approach and consists of eight pipelined modules, such as de-formatter, morphological analyzer, part-of-speech (PoS) tagger, lexical transfer module, structural transfer module, morphological generator, post-generator, and re-formatter.

Google Translator [28] translates a section of a text, or a webpage, into another language. It does not always deliver accurate translations and does not apply grammatical rules, since its algorithms are based on statistical analysis rather than traditional rule-based analysis.

Babel Fish [25] is a web-based application developed by AltaVista, which translates text or web pages from one or several languages into another. The translation technology for Babel Fish is provided by systran [22], whose technology also powers the translator at Google and a number of other sites. It can translate among English, simplified Chinese, traditional Chinese, Dutch, French, German, Greek, Italian, Japanese, Korean, Portuguese, Russian, and Spanish. A number of sites have sprung up that use the Babel Fish service to translate back and forth between one or more languages.

The Anusaaraka [2] is a popular machine-aided translation system for Indian languages that makes text in one Indian language accessible to another Indian language. Further, this system uses Paninian Grammar model [1] to its language analysis. The Anusaaraka project has been developed to translate Punjabi, Bengali, Telugu, Kannada and Marathi languages into Hindi. The approach and lexicon is general, but the system has mainly been applied for children's stories.

Angalabharti [5][6] is also a human-aided machine translation system used in India. Since India has many languages, there are a variety of machine translation systems. For example, Angalahindi[5] translates English to Hindi using machine-aided translation methodology. Human-aided machine translation approach is a common feature of most Indian machine translation systems. In addition, these systems also use the concepts of both pre-editing and post-editing as the means of human intervention in the machine translation system.

Electronic Dictionary Research (EDR) [36], by the Japanese is the most successful machine translation system. This system has taken a knowledge-based approach in which the translation process is supported by several dictionaries and a huge corpus. While using the knowledge-based approach, EDR is governed by a process of statistical machine translation. When compared with other machine translation systems, EDR is more than a mere translation system and hence provides lots of related information.

Table 1 shows a comparison of some existing machine translation systems.

System	Language pair	Approach & Type
Anusaaraka	Among Indian languages	Human-assisted, application
Angalabarath	English to Indian languages	Human-assisted, rule-based, application
AngalaHindi	English to Hindi	Machine-aid, rule-based/ example-based, web-based
ManTra	English to Hindi	Human-aided, web based
English to Urdu MT	English to Urdu	Example based, application
Matra	English to Hindi	Human-aided, transfer-based application
Google TR	Several languages	Statistical, web-based
Bable fish	Several	Systran technology,

	languages	web based
Yahoo TR	Several languages	Statistical, web-based
Aprtium	Related languages	Rule-based, application
EDR	English/Japanese	Knowledge based, application

Table 1: Comparison of the MT systems

At present there are many Sinhala language resources available; including Sinhala Unicode [26], some bilingual dictionaries [20][21], Sinhala corpus[29], some transliteration and OCR systems. However, only few researches have been done on machine translation. Vitanage's English to Sinhala translator for weather forecasting domain [17] and Silva and others' Sinhala to English language translator [16] are some prototype projects. In addition, there some attempts have been taken to develop Sinhala to Tamil machine translation [18] and Japanese to Sinhala machine translation [15].

It is evident from the discussion that we have developed a English to Sinhala machine translation system (BEES). This system has also taken the approach of human-assisted translation and it works on the concepts of Varanagema in Sinhala language. This system has been tested through several standard desktop applications and a web application. Following section reports previous development of the BEES.

Previous Development of the BEES

Our English to Sinhala machine translation system has been primarily implemented with the use of SWI-Prolog [23], Java and Prolog Server Pages PSP [24]. The core of our MT system has seven modules, namely; English morphological analyzer, English parser, word level translator, Sinhala morphological analyzer, Sinhala parser, transliteration module and lexical dictionaries. Our project has introduced the first ever parser [7] and morphological analyzer [8] for Sinhala language. Figure 1 shows the basic interface of our stand-alone machine translation system. This first version of the BEES can translate only simple present tense sentences. It can handle only simple subject and object forms with adjectives, adverbs and articles. Further, to handle out-of-vocabulary issues, it can transliterate English terms into Sinhala. However, this version does not handle semantic issues.

To improve this basic system we have developed three types of systems namely; Human assisted machine translation system [11], web-based English to Sinhala translation system [12][13][27] and context based English to Sinhala machine translation system[14]. The web-based English to Sinhala translation system is a web-enabled version of the stranded English to Sinhala machine translation system. A brief description of the other two developments is given bellow.

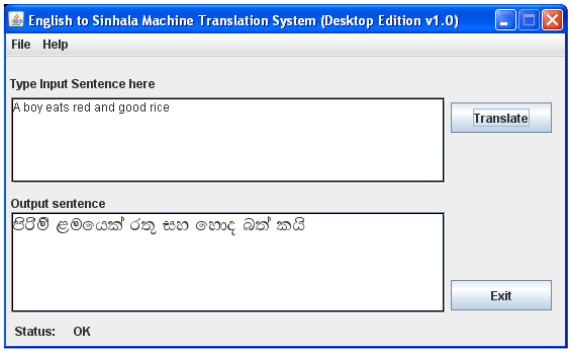


Figure 1: stand-alone Machine translation system

Human-Assisted machine translation system

Human-assisted machine translation system has been developed to solve out-of-vocabulary and semantic issues in the English to Sinhala machine translation. This application has been developed as a java based application and it runs on Linux or Windows based systems. This system provides user interface (Intermediate editor) to semantic handling. Figure 2 shows the user interface of the intermediate editor.

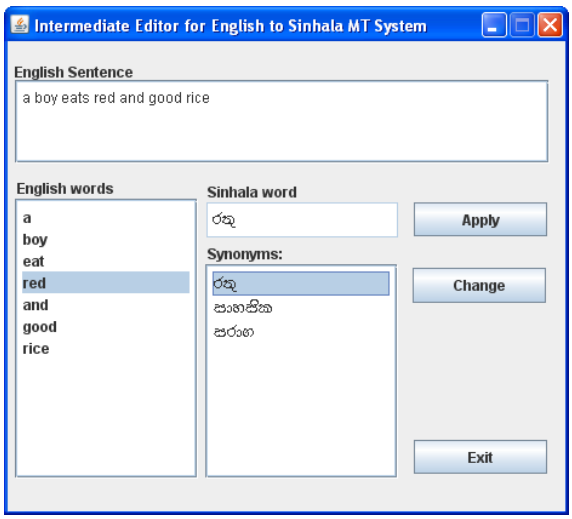


Figure 2: The intermediate editor

This editor provides facilities such as display of synonyms and antonyms and related words. The intermediate-editor is linked with both English and Sinhala dictionaries in the MT system. The process of intermediate-editing, before composing a Sinhala sentence drastically reduces computational costs of running a Sinhala morphological analyzer and parser. In addition, the requirement for post-editing can be reduced by the process of intermediate editing. On the other hand, intermediate-editing can be used as a means of continuous capturing of human expertise for machine translation. This knowledge can be

reused for subsequent translations. With the above ideas we have developed a context-based, English to Sinhala machine translation system to use the human knowledge through the concept dictionary.

Context based handling system

Development of the fully automated, perfectly correct translation system is very difficult for any language pairs. However, we are researching to develop a fully automated machine translation system, using the captured human knowledge throughout the result of the intermediate editing. The result of the intermediate editing is stored in a dictionary named concept dictionary. This information can be used to handle the semantics in the Machine Translation. By using this context-based information, we have developed a context-based machine translation system that translates English paragraphs in to Sinhala. This system has the following features;

- Handling multiple sentences.
- Ability to handle semantics through concept dictionary.
- Ability to handle simple and complex sentences
- Ability to translate all tenses with active and passive

However we have noted that, English to Sinhala web page translation is more useful for many people who use web resources. Therefore, we have developed a new version of English to Sinhala machine translation system that can translate a given English web page into Sinhala. Design of the system is given bellow.

Design of the BEES

The translation system is designed to translate a given English web page into Sinhala. This system contains two modules namely translation module and the HTML parser. Figure 3 shows the overview of the web page translation system.

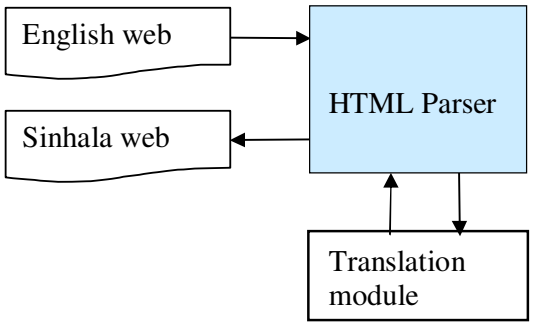


Figure 3: Over view of the translation system

The input of the system is an English web page and the output is a translated Sinhala web page. Brief description of each module is given below.

HTML Parser

The HTML parser is the controlling module of the system. As the first step, the parser analyzes the input HTML document and decodes the text and tags. Then the HTML parser sends the text into the Translation module and gets the Sinhala translated text. Finally, the system composes the web page using these text and tags. The HTML parser has been developed using JAVA.

Translation module (BEES)

We have designed the BEES with seven modules, namely; English morphological analyzer, English parser, word level translator, Sinhala morphological analyzer, Sinhala parser, transliteration module and four lexical dictionaries namely English dictionary, Sinhala dictionary, English to Sinhala bilingual dictionary and concept dictionary. Figure 4 shows the design of the BEES. Note that this new design of the BEES does not contain Inter-mediate editor. This is because this system uses concept dictionary for semantic handling. The concept dictionary is updated through the previous development of the BEES. Brief descriptions of each module are given bellow.

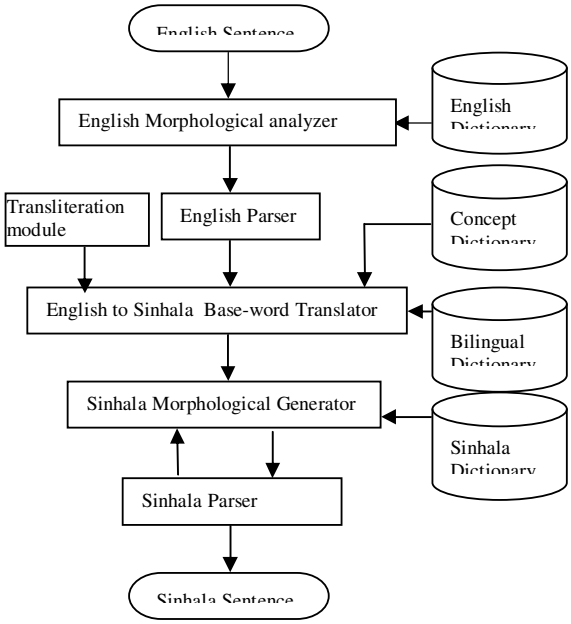


Figure 4: Design of the BEES

English Morphological analyzer reads a given English sentence, word by word and identifies morphological information for each word. The morphological analyzer in our MT system has linked up with an English dictionary to get grammatical information of the words in the input sentence. Using SWI-PROLOG, we have developed a rule based English morphological analyzer for our purpose.

The English parser receives source English sentences and tokens from English Morphological analyzer. This parser works as a syntax analyzer. Since there are many English parsers, we have customized an existing parser for our purpose. The current version of the parser can handle simple and complex sentences including active and passive tenses. The parser has also been implemented using SWI-PROLOG.

The word level translator is used to translate English base-word into Sinhala base-words with the help of the bilingual dictionary and the concept dictionary.

The Sinhala morphological analyzer [7] works as a morphological generator. This morphological analyzer reads the words from the translator word by word. For each word, the morphological analyzer generates the appropriate word with full grammatical information such as nama (nouns), kriya (verb) and nipatha (preposition) in the Sinhala language [31][32]. This analyzer is based on Akshar's and others Morphological Analysis Shell[3] and uses rule based approach for concepts of "Varanagema". It works with the help of two dictionaries, namely, Sinhala rule dictionary and Sinhala word dictionary. All these dictionaries and the Sinhala morphological analyzer have been implemented using Prolog.

The Sinhala parser [6] works as a sentence composer. It receives tokenized words from the Sinhala morphological analyzer and composes grammatically correct Sinhala sentences. In general, a Sinhala sentence contains 5 components, namely Ukktha vishashana (adjunct of subject), Ukkthya (Subject), karma vishashanaya (attributive adjunct of object), karmaya (object) and akkyanaya [33][31]. These five components of a Sinhala sentence are the building blocks for the design and implementation of a Sinhala parser. The parser is also one of the key modules of this English to Sinhala Machine Translation System and it has also been implemented using SWI-PROLOG.

Translation system uses four dictionaries such as English dictionary, English-Sinhala bilingual dictionary, Sinhala dictionary and concept dictionary. The English word dictionary contains English words and the lexical information. English to Sinhala bilingual dictionary is used to identify appropriate Sinhala base word for a given English word and it contains the relation between English and Sinhala words. Sinhala dictionary contains two sub dictionaries namely; Sinhala word dictionary and Sinhala rule dictionary. The Sinhala word dictionary stores Sinhala regular base words and lexical information. The Sinhala rule dictionary stores rules required to generate various word forms. These are the inflection rules for formation of various forms of verbs and nouns from their base words. The rule dictionary also stores vowels, consonants, upasarga (prefix) and vibakthi (postfix). The concept dictionary contains three sub dictionaries namely; English concept dictionary, Sinhala concepts dictionary and bilingual concept dictionary. The English concept dictionary contains synonyms, anti-synonyms and general knowledge about English words. Similar to the English dictionary, Sinhala concept dictionary stores Symantec information. The bilingual concepts dictionary stores bilingual semantic information which are update by humans through the intermediate editing.

Transliteration module is used to solve out-of-vocabulary problems and to translate technical terms. Transliteration is the practice of transcribing a word or text written in one writing system into another writing system [10]. In other words, machine transliteration is a method for automatic conversion of words in one language into phonetically equivalent ones in another language. At present, we have developed two types of transliteration models. One of these models transliterates original English texts into Sinhala Transliteration and the other transliterates Sinhala words that are written in English, which transliterate into Sinhala. Finite state transducers are used to develop these two modules.

The following section reports some translation issues that are handled by the system.

Translation issues

The English to Sinhala web page translation is a critical process considering the large and complex type of sentences. This section describes some common issues that are addressed by the BEES.

Text manipulation issues

An html document contains a lot of tags and text. The text on the web document is not completely sentences. These texts are available in several formats such as;

- Complete sentences
- Noun phrases
- URLs
- Equations
- Numbers etc.

The web translation system needs to handle these texts for target language generation. Identification of the complete sentence is one of the critical problems in the context based machine translation. Any sentence in English ends with a dot sign (.) and after the dot sign the space appears. Using these two character combinations, the system identifies the sentence. However there is a problem in understanding names (Example: A. B. Fernando) Note that, the "A." is not a sentence ending therefore HTML parser uses internal mechanism to remove this issue. Also Noun phrase identification is another issue in the translation. As an example consider the following

phrase "A Computer Science Subject", is translated as a "mrs.Kl jsoHd jsIhla". Note that there are grammatical differences between English and Sinhala languages; therefore, word level translation cannot be used. This is because there is a difference between Sinhala nouns in the noun form and adjective form ("mrs.Klh" is a noun form and mrs.Kl is an adjective form.) [33] Also in Sinhala, article comes with a Sinhala noun. According to the above reasons, we have developed a translation module to translate noun phrases. However, URLs, Numbers and equations cannot be translated.

Grammatical issues

There are several issues that have been addressed by the present system. Due to having different language structures in English and Sinhala languages, the translation of English to Sinhala is a difficult process. English is a West Germanic language that originated in Anglo-Saxon England. Sinhala belongs to the Indo-Aryan branch of the Indo-European languages [29]. Following list shows some grammatical issues in both languages.

- The literary language and the spoken language differ from each other in Sinhala.
- Sinhala uses SOV (Subject Object Verb) word order and English uses SVO (Subject Verb Object) word order.
- Sinhala nouns have five types of inflections, namely, gender, number, person, case and artical (difinite/indifinite). English nouns have four types of inflections, namly; gender, number, person and case.
- Sinhala has nine cases and these differ from English.
- There is a difference between noun and the adjective form of the noun in Sinhla but no such difference is found in English.
- Sinhala language contains only three tenses while English has 12 tenses.
- Sinhala sentences contain 5 components, namely Ukktha vishashana (adjunct of subject), Ukkthya (Subject), karma vishashanaya (attributive adjunct of object), karmaya (object) and akkyanaya. However, this structure is different from the English sentence structure.

This English to Sinhala machine translation system uses the concept dictionary to its semantic handling. The following section shows how the system works for a given input text.

How system work

This section describes how the system translates a given English web page into Sinhala. Figure 5 shows the user interface of the system.

Figure 5: User interface of the BEES

To start the translation, you need to select a web page and click the translation button. After the translation, the system shows the output of the translation by using a web browser. Figure 6 shows the translated output of the Sinhala web page. Assuming that the system reads the following simple HTML document, as the first step HTML parser analyzes the document and identifies the tags and the text. Consider the following simple part of the html document.

```
<tr><td>
        The Rabbit
</td></tr>
<tr><td>
        <img src="trabsl1.jpg">
        The Rabbit is a small and herbivorous animal.
It lives in the jungle. Rabbit has long and powerful legs.
</td></tr>
```

This HTML source contains several HTML tags and text. "The rabbit" is a text identified by the HTML parser. Then the parser sends this text into the translation module. Translation module reads the above text and tries to translate. In the sentence analyzing stage, the English parser rejects the input text, because it is not a sentence. Therefore, the system tries to identify it as a noun phrase. At the moment, the English parser recognized the input text "The rabbit" as a noun phrase. Then the translation module uses the English to Sinhala word translator, Sinhala morphological analyzer and the Sinhala parser, and generates the appropriate Sinhala translation as "ydjd".

This is the time to show how a translation module works for a given complete sentence. Assume that the translation module reads the sentence "The Rabbit is a small and herbivorous

animal" as an input text. Then thw English morphological analyzer reads the input sentence and returns the following.

```
eng_detm([e1000002], dr, 'the').
eng_noun([e1000077], td, sg, ma, sb, 'rabbit').
eng_verb([e1000057], if, 'is').
eng_detm([e1000001], id, 'a').
eng_adjv([e1000074], p, 'small').
eng_conj([e1000020], 0, 'and').
eng_adjv([e1000076], p, 'herbivorous').
eng_noun([e1000059], td, sg, co, sb, 'animal').
```

eng_detm/3, eng_noun/6, eng_verb/3, eng_adjv/3 and eng_conj/3 are the prolog predicates to represent English words. Then English parser reserves above information and analyzes the English sentence. The English parser returns the following predicates.

```
eng_sentence_type(simple,if).
eng_sen_verb([e1000057]).
eng_sen_complement([e1000001, e1000074, …]).
eng_sen_subject([e1000002, e1000077]).
eng_sen_ekeys([e1000002, e1000077, …]).
```

This English parser indentifies the subject, verb and complement of the sentence. It stores these information using prolog predicates such as eng_sen_verb/1, eng_sen_complement/1 and eng_sen_subject/1. After successful syntax analysis, word translator translates the correspondent Sinhala root word for a given input root word. The word translator returns the following predicates.

```
estrwords(1001, e1000002, s1000000, dt).
estrwords(1002, e1000077, s1000078, na).
estrwords(1003, e1000057, s1000059, vb).
estrwords(1004, e1000001, s1000000, dt).
estrwords(1005, e1000074, s1000076, aj).
estrwords(1006, e1000020, s1000018, cn).
estrwords(1007, e1000076, s1000077, aj).
estrwords(1008, e1000059, s1000060, na).
```

The estrwords/4 prolog predicates represent bilingual information for each English root word. By using this information Sinhala morphological generator generates suitable Sinhala words for the corresponding English word with full grammatical information.

```
snoun([s1000078], td, sg, ma, li, dr, v1,'ydjd').
sin_fverb([s1000059], td, sg, pr,'h').
sin_adjv([s1000076],'l=vd').
sin_conj([s1000018],'iy').
sin_adjv([s1000077],'Ydl NlaIl').
snoun([s1000060], td, sg, co, li, id, v1,'isjqmdfjla').
```

Using all these information the Sinhala parser generates the appropriate Sinhala sentence as "ydjd l=vd iy Ydl NlaIl isjqmdfjlah".

After the successful translation HTML parser reads these translated texts and composes a corresponding web page. Using this interface the user can see the original English web page and the translated Sinhala web page separately. Figure 6 shows the output web interface of the translator.

Figure 6: Translated output web page

Conclusion and Further Works

This paper has reported the design and implementation of the English to Sinhala machine translation system that can translate an English web page into Sinhala using the concept of "Varanagema". The "Varanagema" concept has reduced the workload of the Sinhala morphological generation and the number of word forms to be stored in dictionaries. Further the context based approach is used to semantic handling in the system. Therefore this system becomes a fully automated system.

However, we have identified that the identification of the context in the paragraphs or a sentence is a complex task and hence needs improvement. Updating the lexical resources and generating an algorithm to identify the context of the text are further work of this project.

References

[1] Akshar B., Chaitanya V., Sangal R., Natural Language Processing: A Paninian Perspective, Prentice Hall of India, New Delhi, India, 1995.

[2] Akshar Bharati, Vineet Chaitanya, Amba P. Kulkami, & Rajeev Sangal: Anusaaraka: machine translation in stages. Vivek: a

Quarterly in Artificial Intelligence, vol.10, no.3, July 1997, pp.22-25.

[3] Akshar B., Sangal R., Sharma D. M., Mamidi R., Generic Morphological Analysis Shell, In Proceedings of LREC 2004 -SALTMIL Workshop: First Steps in Language Documentation for Minority Languages. Lisbon, Portugal, 2004.

[4] Sinha R.M.K, Jain A., "AnglaHindi: an English to Hindi machine-aided translation system", MT Summit IX, New Orleans, USA, 23-27 September 2003; pp.494-497.

[5] Mahesh R, Sinha K., Integrating CAT and MT in AnglaBhart-II architecture, 10th EAMT conference "Practical applications of machine translation", May 2005, pp235-244.

[6] Hemant D., Computer-assisted translation system – an Indian perspective. Machine Translation Summit VII, 13th-17th September 1999, Kent Ridge Digital Labs, Singapore. Proceedings of MT Summit VII "MT in the Great Translation Era"; pp.80-85.

[7] Hettige B., Karunananda A. S., A Parser for Sinhala Language – First Step Towards English to Sihala Machine Translation, proceedings of International Conference on Industrial and Information Systems (ICIIS2006), IEEE, Sri Lanka, 2006, pp583-587.

[8] Hettige B., Karunananda A. S., A Morphological analyzer to enable English to Sinhala Machine Translation, Proceedings of the 2nd International Conference on Information and Automation (ICIA2006), Colombo, Sri Lanka, 2006, pp. 21-26.

[9] Hettige B., Karunananda A. S., Developing Lexicon Databases for English to Sinhala Machine Translation, proceedings of second International Conference on Industrial and Information Systems (ICIIS2007), IEEE, Sri Lanka, 2007, pp. 215-220.

[10] Hettige B., Karunananda A. S., Transliteration System for English to Sinhala Machine Translation, proceedings of second International Conference on Industrial and Information Systems (ICIIS2007), IEEE, Sri Lanka, 2007, pp.209-214.

[11] Hettige B., Karunananda A. S., Using Computer-Assisted Machine Translation to overcome language barrier in Sri Lanka, Proceedings of the 4th Annual Sessions of Sri Lanka Association for Artificial Intelligence (SLAAI), University of Moratuwa, 2007.

[12] Hettige B. Karunananda A.S., Web-based English-Sinhala translator in action, Proceedings of the 4th International conference on Information and Automation foe Sustainability (ICIAfS 08), IEEE, Sri Lanka 2008, pp 80-85.

[13] Hettige B. Karunananda A.S., Web-based English to Sinhala Selected Texts Translation system, Proceedings of the 5th Annual Sessions of Sri Lanka Association for Artificial Intelligence (SLAAI), The Open University of Sri Lanka, October 2008.

[14] Hettige B. Karunananda A.S., Context-Based approach to semantic handling in English to Sinhala Machine Translation, Poster presentation on 27th National Information Techonology conference (NITC09), Sri Lanka, September 2009.

[15] Herath. A , Hyodo. Y, Kawad Y, Ikeda T, A Practical Machine Translation, System from Japanese to Modern Sinhalese, The Logico-Linguistic Society of Japan ,1995.

[16] De Silva, D.; Alahakoon, A.; Udayangani, I.; Kumara, V.; Kolonnage, D.; Perera, H.; Thelijjagoda, S.,Sinhala to English Language Translator, Proceedings of the 4th International conference on Information and Automation foe Sustainability (ICIAfS 08), IEEE, Sri Lanka 2008, pp 419-424.

[17] Vithanage N. V. C. T., English to Sinhala Intelligent Translator for Weather forecasting domain, Thesis submitted BIT degree, University of Colombo, Sri Lanka, 2003.

[18] Weerasinghe R, A Statistical Machine Translation Approach to Sinhala-Tamil Language Translation, Department of Computation and Inteligent Systems, University of Colombo School of Computing, Sri Lanka.

[19] Wikipedia, http://www.wikipedia.org

[20] Madhura dictionary, http://www.maduraonline.com/

[21] Vidudaya dictionay, http://www.dscs.sjp.ac.lk/sinres/index.htm

[22] SYSTRAN : www.systransoft.com/

[23] SWI-PROLOG: http://www.swi-prolog.org

[24] Prolog server page: URL: www.benjaminjohnston.com.au

[25] Babel Fish, http://babelfish.yahoo.com/

[26] Sinhala Unicode, http://locallanguages.lk/

[27] BEES: http://www.dscs.sjp.ac.lk/psp/bees.htm

[28] Google translator, http://translate.google.com

[29] Language Technology Research Laboratory" URL:http://www.ucsc.cmb.ac.lk/ltrl/

[30] Apertium Machine translation system, http://www.apertium.org/

[31] Gunasekara A. M., A Comprehensive Grammar of the Sinhalese Language, Asian Educational Services, New Delhi, Madras, India, 1999.

[32] Karunathilaka W. S., Sinahala Basha Viyakaranaya, M.D. Gunaseena & Company, Clolombo 11, Sri Lanka, 2003.

[33] Karunarathna S., "Sinahala Viharanaya", Washana prakasakayo, Dankotuwa, Sri Lanka, 2004.

[34] Wren P.C., Martin H., High School English grammar and Composition, S. Chand and Company Ltd, Ram Nagar, New Delhi, India, 2005.

[35] Tafseer A., Sadaf A., English To Urdu Translation System, University of Karachi, 2002.
URL:
http://www.khazina.org/files/translator02.pdf

[36] Toshio Y, "The EDR electronic dictionary", Communications of the ACM, Volume 38, Issue 11, 1995, pp. 42 – 44.

SLR Navigator

K.G.Nishadi Lakshika, S.D.H.S.Wikramarathna
G.T.I. Karunarathna, Dr.J.C. Balasuriya, Prof. A.S.Karunananda

Faculty of Information Technology, University of Moratuwa, Sri Lanka
Email: harshisd@gmail.com, nlakshika@gmail.com

Abstract: **SLR Navigator which is an acronym for Sri Lankan Railway Navigator is a multi agent system which has been designed to handle the train delays and avoid unpredictable train crashes in a dynamic situation as a human thinkable train scheduling system. The system has been implemented to overcome the current issues in the manual train scheduling system. SLR Navigator consists of two modules; In Motion Train Tracking System (module 1) and Intelligent Train Control System (module 2).**
The system has been implemented as a Multi Agent System so that it can have all the strengths provided by MAS technologies. Each and every agent can have an idea about the overall situation by reading the message space continuously. They can then contribute to the decision making process when necessary. A two way communication will be established among trains and signal towers as trains may also need to pass messages to the control agent in a panic situation. The ultimate output of the system will be a proper train scheduling process which can smoothly handle train delays and avoid train crashes.

1.0 INTRODUCTION

Sri Lankan Railway is providing travelling services to their customers and it is one of the major infrastructures which directly affects the development process of Sri Lanka. A major part of the work force of the country is using the Railway for their traveling purposes and it highlights the need of efficient and effective service from the railway [12]. Therefore, to improve the service, they need to have a good decision making system and also they should maintain a proper communication with the trains which are on the move. In the existing system of the railway department, these functions are done manually. They maintain traveling information in large excel sheets and take decisions according to those information [12]. The decision making process is highly centralized and as a result it is very difficult to react accurately in an emergency. There is no method to have a direct communication with the train in between stations. Therefore, the maintenance of this system is so difficult and can depend on incorrect information.
The project SLR Navigator gives solutions to these problems by replacing both traditional communicating system and the decision making system of the Railway with newly created two modules of In Motion Train Tracking System and Intelligent Train Controlling System. The target of the module one is to track the current location and the speed of ongoing trains and to establish a proper communication with them. Module two will take needed information from module one and will take necessary decisions about delayed and unscheduled trains. Those decisions will be passed to the train through the established communicating method.
The module 1 uses Radio Frequency Identification (RFID) technology to detect trains and their speeds at signal towers [3]. Then it will use Global System for Mobile (GSM) technology [4] to send this information to module 2 which use them as the inputs. And also Module 1 communicates with trains and passes messages to the trains regarding their speeds. An LCD display will be fixed to the trains where these messages can be displayed to the drivers. The module 2 maintains the message space which all the agents in the system read and write. The controller agent, which is the heart of the Multi Agent System (MAS) [7] analyzes the train behaviors and takes necessary decisions to manage the critical situations with the aid of Fuzzy Logic [8].

2.0 LITERATURE REVIEW

Doing a literature survey to identify the similar approaches to the problem area was one of the first steps taken in this project. When doing a survey, several approaches using RFID and GSM technology to track the vehicles were found. And also there were several applications which have been developed using Multi Agent Systems and Fuzzy Logic.

2.1 Vehicle Tracking Applications

The focus was directed to some main applications regarding the vehicle tracking. The first one is Falcon EVR (Electronic Vehicle Registration) system. It enables vehicle remote monitoring and centralized fleet management using the radio frequency technology to electronically identify vehicles and validate the identity, status, and authenticity of vehicle data [3].
Next studies have been done on a vehicle tracking method proposed by Savoie and Boulay[11]. This system utilizes the cellular network to locate vehicles by paging a cellular transceiver, which is

installed on the vehicle, to identify which cell sites are near the vehicle [11].

There are some projects which used Global Positioning Systems (GPS) technology to track the vehicles. The solution proposed by Annett and Swarbeick is one such project using GPS to track vehicles [11]. GPS has valuable features, such as the ability to track a vehicle's location continuously, even off-route, and the ability to use that information to determine its speed and direction. However, GPS systems are both expensive and hard to set up for vehicles on a set route.

In Turkey, the municipality of Pendik deployed the vehicle tracking system using RFID to control access to an employee-only lot within a large parking area, and to restrict unauthorized vehicles from entering [10]. There are similar types of projects developed to track vehicles using RFID technology **[2].** One of them is a vehicle tracking system implemented in Venis to recognize the vehicle by using its RFID tag [15].

2.2 MAS Applications related to Train Controlling Systems

In this literature review, it has been studied about several MAS approaches for train controlling. A logic programming based environment for MAS prototyping has been adopted for the management of freight train traffic along the railway line between two Italian stations [1]. The research has successfully demonstrated the advantages of the MAS approach to this field of application.

Railway Intelligent Transportation System (RITS) [5] is also another good example for the related works. It is an intelligent transportation system which aims at the guarantee of safety, the increase of transportation efficiency, the improvement of management, the enhancement of the service quality through optimized deploitation of the RT-related mobile, fixed, spatial, temporal and human resources on basis of information collecting, transferring, processing and sharing.

Another MAS Approach to Train Delay Handling has been developed currently for the Swedish railway for calculating Estimated Time of Arrival – ETA [6]. The system mainly consists of two parts. A railway network simulator based on traditional simulation techniques and a multi-agent based simulator of the decision making actors.

The fuzzy logic control unit built by Hitachi Ltd of Tokyo in 1988 runs subway operations in Sendai, Japan [16]. This automatic train operations system regulates train speed more precisely than the best human motorman and serves as a showcase for fuzzy-logic gear. Fuzzy control computers accelerate and brake trains more smoothly than a human driver. In addition, they can help rearrange departure schedules to compensate delays.

According to the review there are a lot of applications which use these technologies. But it is hard to find any application which gathers the strengths of all these technologies. So the main objective of SLR Navigator is to integrate all these technologies for one better solution.

3.0 SLR NAVIGATOR

3.1 In motion Train Tracking System

In Motion Train Tracking System is the first module of this project. In order to take decisions regarding delayed and unscheduled trains it is essential to identify the trains uniquely. To do that RFID technology is used [2, 13]. There is a RFID reader in each and every signal tower and it is responsible for reading the radio frequency and transfering the information to a processing device, and to a RFID tag. There is a RFID tag inside each train. RFID reader located in every signal tower sends out a signal, which activates RFID tags in the train when it reaches that tower. The transponders then reply with an encoded message, which include the train ID and the speed of the train at that location. This message is received by the transceiver and it will decode it. Then it will be read by the signal tower agent located in that tower and the agent will connect to the message space located in the control room to write those information in it.

Connection between the signal tower agent and the control agent will be established through a GSM modem. GMS modems are located in every signal tower and signal tower agent can also be connected to the control agent though that modem. Because GSM technology[14] is basically made to send and receive data streams between two units like PC's or embedded devices it is the most suitable method to transfer data through the modem. By using this connection, the signal tower agent can write into the message space about the train which just passed that signal tower and also the speed of that train. It has the ability not only to write information into the message space but also to read the message space to learn about the train schedule and delayed trains, signal tower agent use that connection established through the GSM modem.

Decision sent to the train agent from signal tower should be displayed in the train because that should be easily seen by the train pilot. Therefore, embedded system was implemented and through the microcontroller that message will be displayed in the LCD display.

3.2 Intelligent Train Controlling System

The main objective of this module is to implement an Intelligent Train Control System with the aid of MAS technology to advise the trains in a situation where some of them are unable to reach the pre-

defined time schedule or in a situation where there are delayed or broken trains on the route.

This module is based on multi agent systems technology. The control center maintains a message space which will be the heart of the system. At the beginning this message space displays the scheduled time table. Any train or signal tower agent can read this schedule and check whether there is any contradiction with the actual behavior. If so, they can alert the other agents by writing it in to message space. The signal tower agents are supposed to update the message space each time when a train passes across the tower. The train ID, speed, location and time will be include in this message. It can also add some more information like whether the train passes at the scheduled time or whether it is delayed. By reading the message space, the signal tower agent gets to know the time a particular train should pass the tower. Therefore, if the train did not pass the tower in the expected time the agent is supposed to alert the others about it. Another signal tower agent may read this message and update the message space saying that the particular train has reached him at this time. In this way a proper communication can be handled among agents and the most important thing is that the control agent can get the overall idea about the train behaviors by reading the message space.

The control agent can then take decisions about trains. For example, if a particular train is getting delayed due to some technical problem, the control agent can decide to reduce the speed of the train coming behind. The amount of speed which should be reduced depends on the speed of the front train and the distance between two trains. Fuzzy logic has been adopted to calculate the speed of the train coming behind. The inputs for the fuzzy system are the speed of the front train and the distance between the two trains. A set of fuzzy rules should be specified according to the user experience. Some experienced train drivers should be interviewed to define these rules. The output will be the speed of the second train. The decisions which are taken in this way are written in to message space. The control agent should follow a set of pre defined rules in the decision making process. These rules will specify how a train crossing should occur, how the train priority should be addressed, how early a train should be informed to stop, how breaking should be done according to the speed etc. And also it should refer to the ontology which should contain information like train time schedule, the priority list of trains etc. Since all agents are communicating through a common message space, all agents have a clear idea about the overall behavior of the system.

4.0 DESIGN OF SLR NAVIGATOR

In order to visualize the whole system at once, high level architecture of SLR Navigator was drawn.

The architecture diagram shown in Figure 1 consists of four major components named Signal Tower Agent, Train Agent, GSM modem and Control Agent. Inside the train agent there is a RFID tag which is used to identify a particular train and to track its speed. RFID reader was located in the GSM modem and when a train arrives at the signal tower, it reads the RFID tag after establishing a connection. Message space for this Multi Agent System is located in the control room where the control agent is resides. Functions of those identified components can be described as follows.

Signal Agent:

- Rread the message space
- Write message space when a train arrived. [contain the train ID, arrival time, whether it is delayed or not]
- Inform about trains which passed that tower in special situations.

Control Agent:

- Write the scheduled time table
- Read the message space continuously
- Identify delayed trains
- Take corrective decisions according to rules
- Display the decisions in message space

Train Agent:

- Read message space
- Execute decisions related to the agent
- Write in message space [reason for late, condition of the train, condition of the path, request for quick advise]

There are few constrains which have been identified in this design. The trains and signal towers communicate with the control agent through GSM modems. The GSM modems are placed in signal towers. The trains themselves do not contain GSM modems due to cost constrains. Therefore, the trains have to wait until they reach a signal tower to establish a connection. This will not be a serious issue to the system as signal towers themselves can inform the control agent about train delays. And also it can decrease the distance among two signal towers to avoid the train being disconnected for a long time.

The area which a GSM modem can cover, its signal strength, its ability to establish connections with moving trains and the noise disturbances are some other constrains related to the communication process which have to be considered. RFID tags are used to identify trains and their current speed. The

distance it can cover is limited. The speed recognition RFID tags are more expensive. Therefore having one RFID tag per train and one RFID reader per signal tower will be less cost effective in practical implementation.

5.0 IMPLEMENTATION

The first step of implementation is to build the fuzzy inference system to calculate the train speeds. This fuzzy system is implemented to control the movement of two trains in one route. The system has been implemented using Math Lab software. If the train which is in front is getting slow or stoped and the distance between two trains is less, then the train behind should slow down or stop. To connect the fuzzy system to the SLR Navigator, JIntega tool has been used. Therefore, each time the control agent needs to take a decision, it connects to the fuzzy system and enters input values and gets the result. Then the control agent distributes these results to the signal tower agents.

The Multi agent system of SLR Navigator contains three types of agents. First one is the Control Agent. Only one agent of this type resides in the system. The next type is Signal Tower Agents. These agents are created and located at every signal tower. They have been named as 1, 2, and 3 according to the tower location. The other type of agent is the Train Agent. These agents are created and located in each train.

The control Agent is the heart of the SLR Navigator. It takes all the train information from signal towers and makes decisions about train movements. The agent periodically searches for available signal tower agents and asks all those signal tower agents to send information about the trains passed by. After getting the train location and speed information it connects to the fuzzy system and takes the appropriate decision. And then the decision will be sent to the signal tower which the particular train passes next. Then the message will be passed to the train when it is passing the signal tower. The train agent also can pass messages to the control agent in certain situations.

Signal tower agents are responsible for getting the train ID and speeding through the RFID when train reaches that tower and writing those into the message space. So establishing and maintaining the connection between control agent and the other agents through the GSM modem is also a responsibility of the signal tower agent. Not only that Signal tower agents will also read the message space continuously and learn about the train schedule and about the delayed trains and if it has some information about that delayed train that will be written in to the message space for control agents' consideration.

When train agent receives the message from the Signal Tower agent, that message will be displayed in the LCD display unit placed in front of the train pilot. So that train pilot can easily see the desired speed for that train so that he can drive in that speed. Implementation of that display unit was done by using microcontroller. 16f877a PIC was programmed and 16*2 LCD display was connected to display that message.

6.0 DISCUSSION

The project SLR navigator has been designed to handle the train delays and avoid unpredictable train crashes in a dynamic situation as a human thinkable train scheduling system. As railway is one of the major transportation services in Sri Lanka, increasing its efficiency and reliability will affect almost all Sri Lankans. SLR Navigator has the ability to identify the trains uniquely and advise them to behave in unexpected situations. RFID is used to identify the trains and to track its speed. That information will be written in to a message space of the Multi agent system. Connection between the control agent and the other agents is established through a GSM modem. Control agent will take some corrective decisions according to the rules using fuzzy inference system and those decisions will be written in to the message space too. Agents can execute decisions which are related to them and hence train crashes and other disasters will be avoided.

The major challenge of addressing this problem was that it is dealing with a real time system. The trains are used to carry a huge number of passengers. Therefore, if the generated speed results are inaccurate, then it will be a mess. The project's approach to address the train scheduling problem differs from the other approaches because it uses the strength of multiple technologies like MAS [7], fuzzy logic[8], RFID[9, 13] and GSM[14]. Hardly any system has used all these technologies to implement systems.

The project SLR Navigator was tested in laboratory using prototypes. It works accurately and all the expected performances were there. The project will also be tested in VBS2 simulator which is a very rich simulation environment close to the real world. The system should be deployed in an existing railway system but a number of testing processes need to be carried out before the deployment of the project. A proper risk analysis also should be carried out before real time using.

There are some further enhancements which can be done to the SLR Navigator. The first one is to track the trains continuously. Currently the train is tracked when it reaches a signal tower only. The position of the train in between the signal towers is predicted by calculations. If the train can be continuously tracked by using some technology like GPS, then the system will be more accurate.

In this project train agents do not replace the human pilot of the train but it just displays the decisions to the pilot to manage train speed. But the system can be enhanced to give the whole responsibility to the train agent and automate the overall functionality of the Rail Way.

7.0 ACKNOWLEDGEMET

This research study has supported by the undergraduate division of the Faculty of IT and the head of the department of Information technology.

8.0 REFERENCES

[1] Alessio Cuppari, Pier Luigi Guida, Maurizio Martelli, Floriano Zini (1999), *Prototyping Freight Trains Traffic Management*

[2] Elisabeth Iliezudor, Zsolt Kemeny(2006), *The RFID technology and its current applications,* MITIP 2006,ISBN 963 86586 5 7, pp.29-36

[3]*Falcon EVR system for automatic vehicles registration by RFID* [online]. [Accessed in October 2008]. Available from World Wide Web:http://geneko.co.yu/eng/news/novosti_ m2m_evr.html

[4]*GSM Technology* [online]. [Accessed in December 2008]. Available from World Wide Web: http://www.canadiancontent.net /mobile/gsm_technology.php

[5] Jia Limin, Jiang Qiuhuna (2003), *Study of Essential Characteristics of RITS*,IEEE Computer Society

[6] Johanna Tornguist, Paul Davidsson (2004), *A Multi-Agent System Approach to Train Delay Handling*

[7] *Multi Agent Systems* [online]. [Accessed in September 2008]. Available from World Wide Web: http://en.wikipedia.org/ wiki/Multi- agent_system

[8] Owen Bishop (2001), *Fuzzy Logic (2)*, Elektor Electronics

[9]*RFID* [online].[Accessed in November 2008]. Available from World Wide Web: http://www.webopedia.com/DidYouKnow/ Co mputer_Science/2005/rfid.asp

[10]*RFID Journal* [online]. [Accessed in October 2008]. Available from World Wide Web: http://www.rfidjournal.com/ article/articleview/3392/1/1/

[11]*RFID tags to track vehicle* [online]. [Accessed in October 2008]. Available from World Wide Web: http://www.financialexpress.com/news/rfid-tags-along-to-track-your-vehicle/139922/

[12] *Sri Lanka Railways* [online]. [Accessed in September 2008]. Available from World Wide Web: http://www.railway.gov.lk/

[13] Susy d'Hont (2006), *The Cutting Edge of RFID Technology and Applications for Manufacturing and Distribution*, Texas Instrument, TIRIS.

[14] Teppo Koskinen(1998), *Fast GSM, Datalink over 9600,* Helsinki University of Technology

[15] *Venice Tracks Vehicles with RFID* [online]. [Accessed in October 2008]. Available from World Wide Web: http://www.wi-fiplanet.com/columns/article.php/3487046

[16] Yui Naoki (1999), *Subway of Sendai City Trafic Bureau*

Main Organizing Committee

- Prof. A. S. Karunananda
- Dr. Nalin Wickramarachchi
- Dr. Gamini Wijayarathna
- Dr. Kapila Ponnamperuma
- Dr. Uditha Ratnayake
- Ms. Menaka Ranasinghe
- Ms. Chadrika Fernando
- Mr. Budditha Hettige
- Mrs. Nilupa Liyanage
- Mr. Kalana Mendis
- Mr. BLD Senavirathne
- Mr. DMSN Dissanayake

Panel of Judgers – Best AI Project Award

- Dr. Damith Karunarathne
- Dr. TGI Fernando
- Dr. MC Wijegunasekara

Panel of Reviewers

- Dr. Nalin Wickramarachchi
- Dr. ND Kodikara

Special Supports

- **Language editing** : Ms. Umesha Weerakkody
- **Compiling** : Ms. Uditha Jayasinghe
  Ms. MGKT Sammani