UNIVERSITI PUTRA MALAYSIA

IMPROVED RSYNC ALGORITHM TO MINIMIZE COMMUNICATION COST USING MULTI-AGENT SYSTEMS FOR SYNCHRONIZATION IN MULTI-LEARNING MANAGEMENT SYSTEMS

AMIR KOMBO MWINYI

FK 2017 73

IMPROVED RSYNC ALGORITHM TO MINIMIZE COMMUNICATION COST USING MULTI-AGENT SYSTEMS FOR SYNCHRONIZATION IN MULTI-LEARNING MANAGEMENT SYSTEMS

By

AMIR KOMBO MWINYI

Thesis Submitted to the School of Graduate Studies, Universiti Putra Malaysia, in Fulfillment of the Requirements for the Degree of Doctor of Philosophy

May 2017

DEDICATION

I dedicate this workd to my beloved mother, Fatma Salim Ahmed who devoted her valuable time to consistently pray and encouraging, as well as support and sacrifice, which will always be remembered but can never be repaid. To my people of my life, my wonderful wife, Nawal Seif Kassim, who has encouraged and supported me so much over the years, and to our lovely children, Asya Amir, and Akram Amir. To all family members for their support and unending encouragement in my entire endeavour.

Abstract of thesis presented to the Senate of Universiti Putra Malaysia in fulfillment of the requirement for the Degree of Doctor of Philosophy

IMPROVED RSYNC ALGORITHM TO MINIMIZE COMMUNICATION COST USING MULTI-AGENT SYSTEMS FOR SYNCHRONIZATION IN MULTI-LEARNING MANAGEMENT SYSTEMS

By

AMIR KOMBO MWINYI

May 2017

Chairman : Associate Professor Syed Abdul Rahman Al-Haddad, PhD
Faculty : Engineering

Remote file synchronization (RFS) is based on updating the outdated version of a file that resides on one machine to be similar to the new version of the updated file in another machine at a minimum computation time (cost). The problem of rsync algorithm synchronization process is that rsync tries to check folders and files one by one, which takes long time to synchronize them. Therefore, the aim of this study is to minimize the computation time during RFS by improving the standard rsync algorithm.

Previously, several algorithms and techniques have been proposed for partial file synchronization but many of them were based on controlling the block size, checksums, and delta compression of the matched blocks, to solve the searching problem of the rsync algorithm. This study proposed several techniques to improve rsync (irsync) algorithm in order to reduce computation time during RFS, by encompasses a Multi-Agent system (MAS) framework. This algorithm involves several agents, such as: initiator, sense_agent (SA), log_agent (LA), and search_agent (SeA). These types of agents have different capabilities, actions, and efficiency to the irsync algorithm in file synchronization.

The study proposed MAS framework in the Learning Management System (LMS) that involves the transfer of data from one machine to another. To meet this requirement, a new Multi LMS (MLMS) model using Sharable Content Object Reference Model (SCORM) specifications to share learning materials among different higher learning institutions (HLIs) has been presented. This model enhances the interoperability and collaboration of HLIs in terms of synchronization and sharing of learning contents. To evaluate the computation time of the new techniques, standard datasets, which include two versions of source codes emacs-19.28 with emacs-19.29, and gcc-4.8.1

with gcc-4.8.2, were used. The experimental results show that the improved rsync (irsync) algorithm yields a better performance against two previous algorithms, rsync algorithm and hierarchical folder synchronization algorithm (HFSA) in terms of reducing computation time and improve synchronization response time.

Therefore, the MAS framework was performed and the reduction of computation time was obtained by 19.86% compared to 42.25% of standard rsync. The results also indicated that reducing the searching time could enhance the irsync algorithm responsiveness time by 32.07% compared to 67.93% of the standard rsync. The integration of the proposed MLMS model with irsync algorithm was further tested through a prototype with MAS and show significant improvement over the cloud synchronization system which based on CDMI technology. These two systems were evaluated in terms of synchronization rate. The results revealed that the MLMS system with irsync (MAS) outperformed the cloud system.

Abstrak tesis dikemukakan kepada Senat Universiti Putra Malaysia sebagai memenuhi keperluan untuk Ijazah Doktor Falsafah

PENAMBAHBAIKAN ALGORITHMA RSYNC UNTUK MEMINIMAKAN KOS KOMUNIKASI DENGAN MENGGUNAKAN SSTEM AGEN PELBAGAI BAGI PENYEGERAKAN DALAM SISTEM PENGURUSAN PEMBELAJARAN PELBAGAI

Oleh

AMIR KOMBO MWINYI

Mei 2017

Pengerusi : Profesor Madya Syed Abdul Rahman Al-Haddad, PhD
Fakulti : Kejuruteraan

Penyelarasan fail secara jauh (RFS) adalah berdasarkan kepada bagaimana untuk mengemas kini versi fail lama yang berada di dalam satu mesin untuk menjadi serupa dengan versi baru bagi fail yang dikemaskini dalam mesin lain pada kos masa yang minimum untuk pengiraan (kos). Masalah proses penyelarasan algoritma ialah rsync cuba untuk memeriksa folder dan fail secara individu, ini mengambil masa yang lama untuk diserak . Oleh itu, tujuan kajian ini adalah untuk mengurangkan kos masa pengiraan semasa RFS dengan menambah baik algorthma arsync piawaian

Sebelum ini, beberapa algoritma dan teknik telah dicadangkan untuk penyelarasan fail separa tetapi kebanyakan mereka adalah berdasarkan kepada mengawal saiz blok, semak jumlah, dan mampatan delta blok sepadan, untuk menyelesaikan masalah pencarian algoritma rsync. Kajian ini mencadangkan beberapa teknik untuk meningkatkan algoritma rsync (irsync) bagi mengurangkan kos masa pengiraan semasa RFS dengan merangkumi rangka kerja sistem multi-ejen (MAS). Algoritma ini melibatkan beberapa ejen, seperti: pemula, sense_agent (SA), log_agen (LA), dan search_agent (SEA). Jenis-jenis agen mempunyai keupayaan, tindakan dan keberkesanan algoritma irsync yang berbeza dalam penyerakan fail.

Kajian ini mencadangkan rangka kerja MAS dalam Sistem Pengurusan Pembelajaran (LMS) yang melibatkan yang melibatkan pemindahan data daripada satu mesin ke mesin yang lain.Bagi memenuhi keperluan ini, satu model baru LMS pelbagai (MLMS) menggunakan spesifikasi Model Pengkongsian Kandungan Objek Rujukan (SCORM) untuk berkongsi bahan pembelajaran dalam kalangan institusi pengajian tinggi yang berbeza telah dikemukakan.Model ini akan meningkatkan daya salingkendali dan kerjasama institusi pengajian tinggi (HLI) dari segi penyelarasan dan perkongsian kandungan pembelajaran.Untuk menilai keberkesanan teknik baru, dataset piawaian, termasuk dua versi kod sumber emacs-19.28 dengan emacs-19.29 , dan gcc-4.8.1 dengan gcc-4.8.2 telah digunakan.Keputusan eksperimen menunjukkan bahawa algoritma rsync (irsync) yang bertambah baik menghasilkan prestasi yang lebih baik berbanding dua algoritma sebelumnya, algoritma rsync dan algoritma folder hierarki penyelarasan (HFSA) dari segi mengurangkan kos masa pengiraan dan penjimatan masa tindak balas penyelarasan.

Oleh itu, rangka kerja MAS telah dijalankan dan pengurangan kos komunikasi tercapai pada 19.86% berbanding dengan 42.25% untuk rsyncpiawaian.Keputusan juga menunjukkan yang mengurangkan masa pencarian boleh menambah baik masa kegerakbalasan algoritma rsync sebanyak 32.0% berbanding dengan 67.93% untuk rsync piawaian. Integrasi model MLMS yang dicadangkan dengan algoritma irsync telah diuji melalui prototaip dengan MAS dan menunjukkan peningkatan yang ketaraberbanding dengan sistem penyelarasan cloud yang dikembangkan yang berdasarkan teknologi CDMI. Kedua-dua sistem-sistem ini telah dinilai dari segi kadar penyelarasan. Keputusan menunjukkan yang sistem MLMS dengan irsync (MAS) lebih baik daripada sistem cloud.

ACKNOWLEDGMENTS

All Praise is due to Almighty ALLAH as He is all Merciful, Most Gracious and Most Compassionate who gathered all knowledge for eternity. May ALLAH praise our Prophet Muhammad (Peace be Upon Him), and render him and his household safe and secure from all evil. All Grace and Thanks belongs to Almighty Allah.

Many special thanks and gratitude go to my supervisor Associate Professor Dr. Syed Abdul Rahman Al Haddad, for his extensive assistance and indispensable comments, advice, and expertise at every stage of this study.

The completion of this research could not have been accomplished without the support of the supervisory committee members, Professor Dr. Rusli bin Hj Abdullah and Associate Professor Dr. Shaiful Jahari Hashim, thank you for your invaluable advices, comments, guidance, and discussions during my study at University Putra Malaysia.

Thanks to every person who has supported me to produce this thesis. I am very grateful to the Faculty of Engineering and the staffs of Computer and Embedded Systems Engineering Laboratory, Postgraduate office, Library, and Universiti Putra Malaysia have provided an excellent environment for my research. Finally, I would also like to extend my sincerely thanks to Zanzibar University in providing me with financial support and time to make this study possible.

This thesis was submitted to the Senate of Universiti Putra Malaysia and has been accepted as fulfillment of the requirement for the degree of Doctor of Philosophy. The members of the Supervisory Committee are as follows:

Syed Abdul Rahman Al Haddad, PhD
Associate Professor
Faculty of Engineering
Universiti Putra Malaysia
(Chairman)

Rusli bin Abdullah, PhD
Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Member)

Shaiful Jahari bin Hashim, PhD
Associate Professor
Faculty of Engineering
Universiti Putra Malaysia
(Member)

ROBIAH BINTI YUNUS, PhD
Professor and Dean
School of Graduate Studies
Universiti Putra Malaysia

Date:

Declaration by Members of Supervisory Committee

This is to confirm that:

- the research conducted and the writing of this thesis was under our supervision;
- supervision responsibilities as stated in the Universiti Putra Malaysia (Graduate Studies) Rules 2003 (Revision 2012-2013) were adhered to.

Signature: ____

Name of Chairman of Supervisory Committee: Associate Professor Dr. Syed Abdul Rahman Al Haddad

Signature: ____

Name of Member of Supervisory Committee: Professor Dr. Rusli bin Abdullah

Signature: ____

Name of Member of Supervisory Committee: Associate Professor Dr. Shaiful Jahari bin Hashim

TABLE OF CONTENTS

LIST OF TABLES

Table **Page**

LIST OF FIGURES

LIST OF ABBREVIATIONS

ACL	Agent Communication Languages
AOM	Agent-Oriented Methodologies
AOP	Agent-Oriented Programming
CDMI	Cloud Data Management Interface
DAD	Detailed Agent Description
FIPA	Foundation for Intelligent Physical
GOD	Goal Overview Diagram
HFSA	Hierarchical Folder Synchronization Algorithm
MLMS	Multi Learning Management System
HPR	Human Practical Reasoning
KQML	Knowledge Query and Manipulation Language
LMS	Learning Management System
MAAD	Multi-Agent Architectural Description
MAS	Multi-Agent System
MRP	Multi-Round Protocol
PDT	Prometheus Design Tool
RFS	Remote File Synchronization
SCORM	Sharable Content Object Reference Model
SRP	Single Round Protocol
SSD	System Structural Description
SUZA	State University of Zanzibar

CHAPTER 1

INTRODUCTION

1.1 Background

Learning management system (LMS) also called Virtual Learning Environment (VLE) is a software application that is focused particularly on teaching and learning that provides course management, assessment, communication, collaboration, curriculum development, and task management automation (Chung, Pasquini, and Koh, 2013). Learners' can log on and work from anywhere around the world where Internet is available, at any time.

To realize the benefits of LMS, a growing number of institutions and business organization have embraced this concept of web based learning systems to fulfil their educational needs (Pavan, Santhi, and Jaisankar, 2012; Hejazinia and Razzazi, 2010; Xiang, 2008). Like other information technology innovation in recent years, the LMS is capable of adding a level of efficiency to institutions' or organizations' learning systems, with several number of benefits, such as: flexible learning times, distance education, manage, and delivering of digital learning contents to the students.

There are a number of software utilities such as Microsoft's AciveSync, Puma Technologies' IntellSync, GoodSync, Dropbox, SmartSync, and rsync that allows synchronization between desktop computers, mobile devices, or web accessible tousers' accounts (Shaoyan, 2009; Agarwal, 2002; Hansen, 2010; Drago, 2012; Gonçalves, 2014; and Kijewski-Correa, 2012). In this scenario, files or records can be updated by several different parties, and size of the file and time stamps can be used to determine which version on which device is the most recent. Synchronization can be used for remote backup of data sets that may have only changed slightly between backups (Tridgell, 2000). In this case, the overhead of keeping the old version at the server is usually unreasonable. File synchronization has also been considered for efficient HTTP transfer between clients and a server (Tridgell, 2000). The advantage is that the server does not have to keeptrack of the old versions held by the clients, and does not need to fetch such versions from disk upon a request.

The RFS is a data transfer method where the sender does not have knowledge about the files held by the receiver in the communication system. Consider the following scenario, Sender S has a copy of file (*string*) f_{new} and Receiver R has an old copy of file f_{old}, the challenge is to design an algorithm that is capable of enables Receiver R to have a copy of file f_{new} while minimizing the communication cost between the two sides. In this case, Sender S will only sends parts of the file that do not match the Receiver R, and hence minimize the communication cost of sending the whole file to the receiver which in turn increase performance of the communication system, particularly in a slow communication link.

Because of the distributed nature of the source code and mirroring of web pages in several servers, synchronization must concurrently consider the communication cost and time taken to transfer changes over the network to achieve high performance. These changes in most cases are updates where an old version of the file is already presented at the source side (Tridgell, 2000). One issue of the file synchronization is the development of effective algorithm that is capable of reducing communication cost in order to reduce computation time. The main advantage of rsync algorithm (Tridgell and MacKerras, 1996) is the ability of utilizing the existing file (at the target device) and updating it remotely without having prior information of the relative states of the files at the other end of the link. The main idea of rsync is to transfer only changes that are not the same between old file and new file held in different computers. This means that instead of sending the compressed (*diff*) file from source to target like other methods of transferring file, such as remote copy (*rcp*) command and file transfer protocol (*ftp*), rsync compares two files remotely by using two types of signatures (weak and strong) from the sender (source) to the receiver (target). The algorithm then sends only those parts of the file that are not matched the receiver file. The difficult part of RFS algorithm is that the sender does not have a copy of the files that held by the receiver. Rsync tool is a famous open source utility used by UNIX distributions that uses rsync algorithm to sync files and folders from one location to another.

This study focuses on the file synchronization and implementation of new techniques for rsync optimization. The second part of this thesis based on designing of interoperability model of multi learning management system (MLMS). This model is then used in the implementation of MLMS prototype that takes advantage of the improved rsync algorithm developed in this research project. This study also investigates the usability of improved rsync algorithm in the collaboration of multiple learning management system (LMS), so that to minimize the overall data transfer time.

1.1.1 Issues in new Synchronization Algorithm

The system design requirements for improving rsync algorithm and the new interoperability multi learning management system using agents is divided into functional and non-functional requirements. The functional requirements specify what this new interoperability learning system should do, while the non-functional requirements specify how this collaborative system should behave. In this new system the following functionalities have been identified:

a) **Searching**: The agents should be able to search a similar file at the receiver device and should provide a high speed searching process. This can be achieved by collaboration of more than one agent and the awareness of the current situation of the environment.
b) **Autonomy**: This property means that agents can work independently and make their own decisions. When consider a system consisting of more than one agent, then the autonomous characteristic of the agent required that the system should be decentralised. Therefore, the responsibility of controlling the agent’ action should performed by the agents themselves.

c) **Adaptiveness:** Agents must be able to continuously adapt to their dynamic environment in order to provide up-to-date learning contents, and determine what to do next to maintain it. This is important, because many scenarios are characterised by dynamism and inherent uncertainty. Consequently, agents will have limited knowledge of the prevalent conditions before deployment. Adaptive agents are capable of continuously adjusting a MLMS model of their environment to best reflect the information that has been gathered so far, and are able to predict and evaluate the outcomes of future decisions.

d) **Performance**: An improved rsync algorithm should ideally be able to give guarantees on the searching speed and interoperability collaboration provided by the MAS. Whereas potential empirical results might be sufficient reason to adopt an algorithm in learning web domains with synchronization capability, and other applications that needs to send only portion of the file instead of transferring the whole file.

1.1.2 Multi-Agent Systems (MAS)

Basically, MAS contains multiagents to achieve certain goals or responsibilities (De Oliveira *et al.*, 2006). In addition, MAS as proposed in various applications is a promising technology paradigm that comprises attractive characteristics like autonomous, intelligent and proactive which can be applied in software engineering and other disciplines for developing various systems of different complexity or applications (Zambonelli *et al.*, 2003; Gunasekera et al., 2013). In order to show the importance of MAS, Talib *et al*. (2012) presented the security issue in cloud computing by incorporating MAS for securing Cloud Data. They introduced a new security access control formula called Formula-Based Cloud Data Access Control. MAS's architecture that was presented consisted of two types of agents: Cloud Service Provider Agent which could provide access to the cloud resources and Cloud Data Confidentiality Agent which was responsible for formulating new access control for Cloud Data Storage.

Furthermore, Ogunnusi and Razak (2013) have introduced a fault-tolerant distribution security protocol for distributed mobile agents which are part of MAS entities to reduce the network intrusion attacks. They utilized wireless local area network (WLAN) environment to detect the intrusive packets in the domain. This new distribution protocol for attack detection has included various agents: Mobile agent, agent server and backup agent server. All these agents are working collaboratively with other components like, certification authority, security domain, messaging system, execution platform and keystore in order to ensure security of collaborating mobile agents from any possible attacks while migrating to the execution platform.

For MAS to work efficiently in any platform there must be a communication and sharing of knowledge to accomplish their goals and tasks. When the agents collaborate and share knowledge, they do so within a group of agents with different capabilities to solve a set of problems. Nor et al. (2009) mentioned three types of knowledge: Organizational knowledge, managerial knowledge and technical knowledge that can be incorporated into the groups in the society or community to share information. MAS have been shown to contain different agents with different capabilities to achieve flexibility and enhance the interaction of system. For the purpose of this study, MAS can be used to facilitate the searching process in the file synchronization in order to reduce the communication overhead.

MAS can offer various means of collaboration (Wooldridge and Jennings, 1995), among different types of agents in the system. Agents are capable of collaborating with other agents not only for exchanging data, but also for helping one another to perform certain tasks. This can be seen also from the model developed by Perez and Uresti (2014), where a number of agents are working together to predict the opponent next move. The model has been illustrated by an experiment with the RoboCup 2D Soccer Simulator. In this study, the agents used had to collaboratively make decisions based on the course availability in the multiple LMS system. In addition, for knowledge sharing and interaction, MAS has to work together to find solutions or solve social or business problems, as individual agents have incomplete capabilities to solve complex problems (Sajja, 2008). Therefore, there is a need for MAS to work together and to share information that maximum performance of the system can be attained.

Boulaalam et al. (2013) have conducted a research that involves mobile agents to accelerate the new product development process. In their study that based on Auto ID, they have incorporated mobile multi-agent system technologies, to improve the innovation process in the enterprise. In their proposed architecture of intelligent product, innovation can be improved by introducing the new product generation by utilizing MAS before the end of the ex-product version.

1.1.3 Multi Learning Management System (MLMS)

Currently, there are a number of LMSs which categorized as open source and proprietary supported LMSs. Many of the large vendors, including Blackboard, Desire2Learn, and eCollege fall into the proprietary supported software systems category. As a result, they have a technical development team and an experienced support services staff prepared to help their users on implementation, troubleshooting, and use their products. All of these companies are continuously updating their products and partnering with other companies as new tools become available. When integration is required with other software systems that is beyond the particular technical team expertise, these vendors have a dedicated professional services groups available to address their specific concerns.

Recently, efforts have been made to design online systems that provide online courses, which encompass multiple e-learning systems. But more sophisticated system which has the ability to dynamically adapt the LMS and to present instructional learning materials according to the learner’s needs is needed to support shareable and collaborative learning environments (Lee and Chong, 2003). According to Lee and

Chong (2003), a first priority should be given to an intermediary system that the main goal is to integrate distributed LMSs to facilitate information sharing and collaborative learning environment between the shared parts.

To show the importance of integrating more than one LMS, Lee and Chong proposed a model which taking advantage of middleware framework, that enable the development of instructional materials that can be able to utilize the distributed *heterogeneous learning sources*. The middleware system that developed by Lee and Chong (2003) was capable of providing sharing and redistribution of learning objects, and carry out collective learning across virtual learning community called Community-Based Learning (CoBL). In this framework, Lee and Chong have used MAS to manage the learners in the community and utilize of shared data model for integrating multi community. Figure 1.1 illustrates the educational middleware which is applied in CoBL.

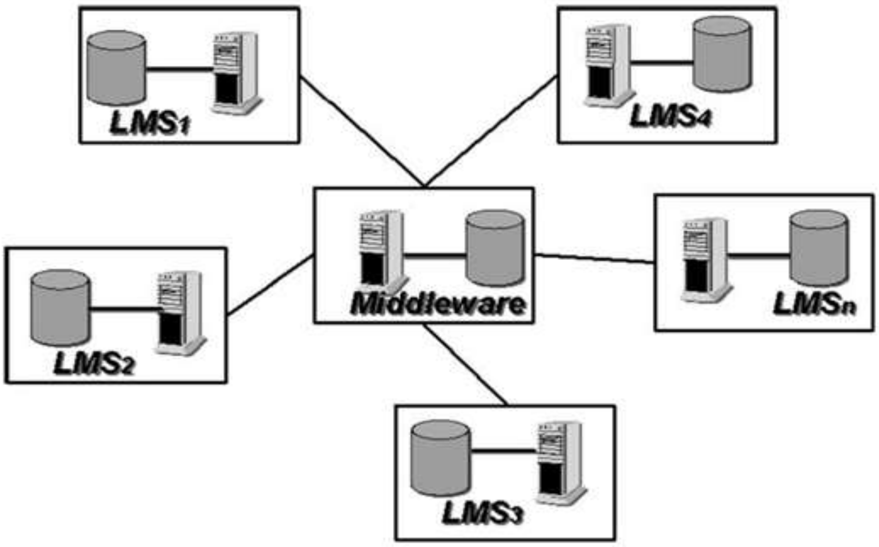

Figure 1.1 : Interacting middleware in CoBL (Lee and Chong, 2003)

The CoBL framework makes use of three advanced technologies (i.e. agent-based communication, component-oriented development, and shared data model). The agent paradigm is employed in the system architecture in order to support communication. The component-oriented development technology allows the design of learning support systems that are context-independent. Shared data model was used to integrating heterogeneous communities, MLMS in our case.

A framework which plays very important part of the community learning (CoBL) was developed using component-oriented. The framework maintains all activities needed for CoBL, also general patterns of the CoBL, which describes state, behaviour, and constraints of learning contents. The CoBL framework manages learner profiles and pedagogical patterns by using rules and constraints in order to determine which learning pattern is best for particular learning conditions (Lee and Chong, 2003). As part of its functionalities, the CoBL framework is required to work with different learners and heterogeneous learning materials in a diverse learning environment. The CoBL framework was built on small components that can be to communicate and collaborate using well-defined interfaces.

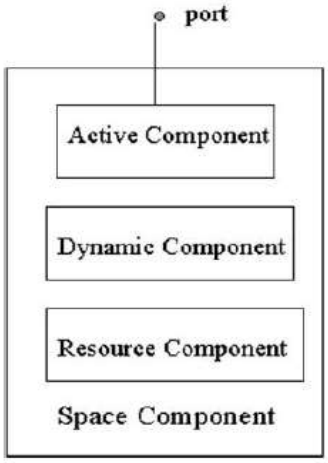

Figure 1.2 : Space component (Lee and Chong, 2003)

The fundamental functionality of the CoBL framework is divided into four different types of components: first is space components, second is active components (using agents to support communications among LMSs), third is resource components (including knowledge bases, and databases), and fourth is dynamic components (i.e. runtime environments). Each component plays a very vital role in supporting the framework. Space components include their own active components, resource components and dynamic components (Figure 1.2). In addition, the resource components keep student (learner) knowledge and lesson. Moreover, the active components (agents) have very important features that giving them the capabilities like communication, autonomous, and proactive which enable them to work in the dynamic environment (Dynamic component). These features facilitate the decisions making that are necessary to create dynamic views. The combination of those four components form improved, adaptive and collaborative LMSs (Lee and Chong, 2003).

1.2 Problem Statement

The main concept of file synchronization is to minimize the amount of data transmitted from the constrained devices. The concept of file synchronization is very simple and easy to understand. File synchronization operates in the situation where two devices held similar files in which it checks the differences between two devices in order to avoid unnecessary transfer of data that already resides in both devices. Therefore, synchronization normally updates both data files by transferring only changes that unmatched during the synchronization process.

There are various algorithms that have been developed to solve synchronization problem, where synchronization of two devices occurs remotely in which both devices do not held the old versions of the data sources. This challenge comes when there are two versions of files in two computers, one is updated and other is outdated. The question is how can we update the outdated version in a minimum computation time in terms of reducing transfer computation time? This will be achieved by reducing RFS computation time during file transfer and file searching process (Ghobadi et al., 2011; Qian et al., 2013; Yang et al., 2014; Sala et al., 2016). According to Ghobadi et al., rsync tried to check folders and files one by one during file synchronization which takes high amount of time to update old files or folders at receiver sides.

Currently, there are many algorithms and tools to solve synchronization problem between different devices. Some of them transfer the entire file if a change has occurred from the updated version, example ActiveSync, and HotSync (Chang, 2009, Schoeny et al., 2014). But if the synchronization involves transfer of huge amount of data between two devices, the transfer of entire file is not sufficient (Ghobadi et al., 2011; Bitouze and Dolecek, 2013; Schoeny et al., 2014; Yang et al., 2014; Sala et al., 2016).

The main gap in the above and other previous literatures in the RFS is that, there is a lack of using multi-agent approach to reduce the computation time in file synchronization. The MAS have shown optimum performance in various applications (Maalal and Addo, 2012; Wang and Botea, 2008; Frankovič and Oravec, 2005; Jansen and Pooch, 2004; Pierre, Hoang, and Pelletier, 2003). Also MAS has been used in conjuction with MD5 to improve file synchronization across different machines in the system called multisync (Niazi et al., 2005). Niazi et al., (2005) they presented an application (Multisync) that utilized mobile agents in a multi-agent system for synchronizing user files seamlessly without user intervention. Another studies that involved MAS and shown good performance in file synchronization were conducted by Manzoor and Ijaz, (2008); Kim et al., (2013); and Kotulski (2015). Again, MAS based system has been used to reduce communication effort between the neighbor subsystems using event-based synchronization (Demir and Lunze, 2012).

This study also focuses on designing an interoperability MLMS model that utilizes an improved rsync algorithm that developed in this study in order to solve the interoperability issue (Leal, and Queiros, 2011) in distributed learning systems. Martínez and Navarra (2007) defined interoperability as “the ability of different computer systems, applications or services to communicate, share and exchange data, information and knowledge in a precise, effective and consistent way”. The improved rsync has been used in interoperability MLMS model to support the synchronization among MLMS so that students can be able to receive updated file automatically from any LMS. The significance of interoperability and collaboration of educational systems is very critical in today learning process (Leal and Queirós, 2011). According to Leal and Queros (2011), currently there is a difficulty to reuse courses in learning institutions with LMS from different vendors; this is a suitable example of the problem found in the majority of the LMSs. Universities in HLIs particularly in Malaysia each university use its own LMS where by no collaboration and sharing of courses with other LMS in other universities are offered (Embi, Hamat, Sulaiman, 2011).

As another objective of this study, the applicability of this model has been tested in the prototype that developed in this study which utilized this proposed interoperability MLMS model. This prototype has been tested and evaluated by comparing it with cloud synchronization system that developed by Yang et al., (2014).

1.3 Research Objectives

The main objective of this research project is to design and implement the new algorithm which will reduce file computation time of searching outdated files during remote file synchronization to enhance rsync algorithm.

Specific objectives of this research were:

i. To design MAS framework with irsync algorithm for reducing the computation time (cost) of the rsync algorithm. The reduction of computation time in irsync is based on the capability of agents to detect changes and transferred only modified (changed) in the file.
ii. To create interoperability MLMS model that utilizes the irsync algorithm to facilitate learning contents synchronization and collaboration in the MLMS. The potential techniques (irsync and MAS) of the interoperability MLMS model compared with the rsync in terms of response time.
iii. To develop and evaluate MLMS prototype that took advantage of the interoperability MLMS model to synchronize learning contents in different courses in MLMS. The prototype has been tested and compared with CDMI in terms of synchronization rate.

1.4 Research Scope

Synchronization and learning management systems are very gigantic research areas, covering computer communication, data compression, data similarity, resource utilization, e-learning, knowledge management, database management, and information sharing. In this study, the main focal point was RFS where the receiver has no prior knowledge of the information stored in the sender, and the development of interoperability model of multiple LMSs that enables the sharing of learning contents (courses). Therefore, network protocols, concurrent synchronization, and other LMS functionalities are not covered in this research. In order to measure the performance of the new algorithm, remote synchronization computation time, communication time caused by agents, and MAS responsiveness in RFS were considered. In terms of interoperability model of MLMS, prototype was developed and compared with the Cloud storage synchronization. The computation time of the irsync and interoperability of the model were quantified. The comparison of these two systems (with and without agents) was evaluated.

To measure the performance of irsync algorithm, a simulation is done using two versions of standard data sources stored in two computers. The performance of irsyc was measured based on the computation time, response time (responsiveness), and communication time by comparing it with standard rsync algorithm and HFSA algorithm. It is also considered that all necessary information for sharing data are ready to be used in all MLMS. Therefore, for the interoperability MLMS model, the objective is to develop a model that integrates multiple LMSs to improve learning process.

In the interoperability MLMS model, the main e-learning standard compliance with combination of MAS with Sharable Content Object Reference Model (SCORM) which facilitates the sharing of learning contents among MLMS.

1.5 Thesis Organization

This thesis presents the rsync algorithm improvement, modeling and development performance evaluation of collaboration MLMS. The remainder of this thesis is organized as follows:

In Chapter 2, the study discusses previous related works. A number of previous techniques are reviewed thoroughly. Based on this review, different techniques used in file synchronization and rsync in particular are reviewed in order to analyze the state of the art. In this chapter, key methods has been identified and adopted with some modification in this research.

Chapter 3, this chapter covers the methodology used in this research. A new algorithm that encompasses intelligent agents to improve rsync algorithm is proposed. In addition, this chapter also presents a generic model for the problem of interoperability in MLMS. Furthermore, system requirements specifications and the basic behaviors of each agent are theoretically described.

Chapter 4 presents the results and discussions of the improved algorithm and collaboration MLMS prototype. In this chapter, performance of the improved rsync algorithm was discussed and compared against standard rsync. Also the empirical analyses of the prototype evaluations were presented.

Finally, Chapter 5 concludes and describes the research directions for future work to extend the agents applicability in the LMS environment. In this chapter, the contributions of this study were also outlined.

REFERENCES

Agarwal, S., Starobinski, D., & Trachtenberg, A. (2002). On the scalability of data synchronization protocols for PDAs and mobile devices. *IEEE network, 16*(4), 22-28.

Alier, M., Mayol, E., Casañ, M. J., Piguillem, J., Merriman, J. W., Conde, M. Á., ... & Severance, C. (2012). Clustering projects for eLearning interoperability. *Journal of Universal Computer Science*, *18*(1), 106-122.

Al-Khalifa, H. S. (2010). A first step in evaluating the usability of Jusur learning management system. In *The 3rd Annual Forum on e-Learning Excellence in the Middle East*.

Anand S. Rao. AgentSpeak(L): BDI agents speak out in a logical computable language. In Walter Van de Velde and John Perram, editors, Proceedings of the Seventh Workshop on Modelling Autonomous Agents in a Multi-Agent World (MAAMAW'96), 22–25 January, Eindhoven, The Netherlands, number 1038 in Lecture Notes in Artificial Intelligence, pages 42–55, London, 1996.Springer-Verlag.

Balachandran, B. M. (2008, January). Developing intelligent agent applications with JADE and JESS. In *Knowledge-based Intelligent Information and Engineering Systems* (pp. 236-244). Springer Berlin Heidelberg.

Balachandran, B. M., Sharma, D., & Peiris, C. (2012). An Intelligent Agent Framework to Manage Web Services Security. In *Intelligent Decision Technologies* (pp. 367-373). Springer Berlin Heidelberg.

Bauer, B., Müller, J. P., & Odell, J. (2001, May). Agent UML: A formalism for specifying multiagent interaction. In *Agent-oriented software engineering* (Vol. 1957, pp. 91-103). Springer, Berlin

Belanche, D., Casaló, L. V., & Guinalíu, M. (2012). Website usability, consumer satisfaction and the intention to use a website: The moderating effect of perceived risk. *Journal of retailing and consumer services*, *19*(1), 124-132.

Belkacemi, R., & Feliachi, A. (2010). Design and deployment of a Multi-Agent System on a Hardware Prototype. In *Power and Energy Society General Meeting, 2010 IEEE* (pp. 1-7). IEEE.

Bellifemine, F., Caire, G., Poggi, A., & Rimassa, G. (2008). JADE: A software framework for developing multi-agent applications. Lessons learned.*Information and Software Technology*, *50*(1), 10-21.

Benta, K. I., Hoszu, A., Văcariu, L., & Creţ, O. (2009, June). Agent based smart house platform with affective control. In *Proceedings of the 2009 Euro American Conference on Telematics and Information Systems: New Opportunities to increase Digital Citizenship* (p. 18). ACM.

Bergenti, F., & Poggi, A. (2009). Multi-agent systems for e-health: Recent projects and initiatives. In *10th Int. Workshop on Objects and Agents*.

Birukov, A., Blanzieri, E., & Giorgini, P. (2005). Implicit: An agent-based recommendation system for web search. In *Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*(pp. 618-624). ACM.

Bitouze, N., & Dolecek, L. (2013). Synchronization from insertions and deletions under a non-binary, non-uniform source. In *Information Theory Proceedings (ISIT), 2013 IEEE International Symposium on* (pp. 2930-2934). IEEE.

Blandford, A., Keith, S., Connell, I., & Edwards, H. (2004). Analytical usability evaluation for digital libraries: a case study. In *Digital Libraries, 2004. Proceedings of the 2004 Joint ACM/IEEE Conference on* (pp. 27-36). IEEE.

Boissier, O., Bordini, R. H., Hübner, J. F., Ricci, A., & Santi, A. (2013). Multi-agent oriented programming with JaCaMo. *Science of Computer Programming*,*78*(6), 747-761.

Bordini, R. H. (2005). A recent experience in teaching multi-agent systems using Jason. In *Proceedings of the 2nd AAMAS Workshop on Teaching Multi-Agent Systems*.

Bordini, R. H., & Hübner, J. F. (2006). BDI agent programming in AgentSpeak using Jason. In *Computational logic in multi-agent systems* (pp. 143-164). Springer Berlin Heidelberg.

Bordini, R. H., & Hübner, J. F. (2009). Agent-based simulation using BDI programming in Jason. *Multi-agent systems: simulation and applications*, 451-471.

Bordini, R. H., Braubach, L., Dastani, M., El Fallah-Seghrouchni, A., Gomez-Sanz, J. J., Leite, J., ... & Ricci, A. (2006). A survey of programming languages and platforms for multi-agent systems. *Informatica (Slovenia)*, *30*(1), 33-44.

Bordini, R. H., Hübner, J. F., & Wooldridge, M. (2007). *Programming multi-agent systems in AgentSpeak using Jason* (Vol. 8). John Wiley & Sons.

Boulaalam, A., E.H. Nfaoui and O.E. Beqqal, 2013. Intelligent product based on mobile agent to accelerate the new product development process. *J. Comput. Sci.*, 9: 856-865.

Bratman ME (1987) *Intention, Plans, and Practical Reason*. Harvard University Press, Cambridge, MA.

Bratman ME (1990) What is intention? In *Intentions in Communication* (ed. Cohen PR, Morgan JL and Pollack ME), pp. 15−32. The MIT Press, Cambridge, MA.

Bray, T., Paoli, J., Sperberg-McQueen, C. M., Maler, E., & Yergeau, F. (1998). Extensible markup language (XML). *World Wide Web Consortium Recommendation REC-xml-19980210. http://www. w3. org/TR/1998/REC-xml-19980210, 16*.

Bresciani, P., Perini, A., Giorgini, P., Giunchiglia, F., & Mylopoulos, J. (2004). Tropos: An agent-oriented software development methodology. *Autonomous Agents and Multi-Agent Systems*, *8*(3), 203-236.

Campbell, J. P., & Oblinger, D. G. (2007). Top-ten teaching and learning issues, 2007. *Educause Quarterly*, *30*(3), 15.

Castro, J., Giorgini, P., Kolp, M., & Mylopoulos, J. (2005). Tropos: a requirements-driven methodology for agent-oriented software. *Agent-Oriented Methodologies, IDEA Group Publishing*, 20-45.

Cernuzzi, L., Molesini, A., Omicini, A., & Zambonelli, F. (2011). Adaptable multi-agent systems: the case of the gaia methodology. *International Journal of Software Engineering and Knowledge Engineering*, *21*(04), 491-521.

Chang, Y. F., Chen, C. S., & Zhou, H. (2009). Smart phone for mobile commerce. *Computer Standards & Interfaces*, *31*(4), 740-747.

Christopher P., (2013). http://elearningindustry.com/learning-management-systems-comparison-checklist-of-features. Retrieved on March 2015.

Chung, C. H., Pasquini, L. A., & Koh, C. E. (2013). Web-based learning management system considerations for higher education. *Learning and Performance Quarterly*, *1*(4), 24-37.

Ciancarini, P. (1992). Parallel programming with logic languages: a survey *Computer Languages*, *17*(4), 213-239.

Conde, M. Á., García-Peñalvo, F., Alier, M., Mayol, E., & Fernández-Llamas, C. (2014). Implementation and design of a service-based framework to integrate personal and institutional learning environments. *Science of Computer Programming*, *88*, 41-53.

Cook, D. J. (2009). Multi-agent smart environments. *Journal of Ambient Intelligence and Smart Environments*, *1*(1), 51-55.

Coria, J. A. G., Castellanos-Garzón, J. A., & Corchado, J. M. (2014). Intelligent business processes composition based on multi-agent systems. *Expert Systems with Applications*, *41*(4), 1189-1205.

Cossentino, M., Chella, A., Lodato, C., Lopes, S., Ribino, P., & Seidita, V. (2012). A notation for modeling jason-like bdi agents. In *Complex, Intelligent and Software Intensive Systems (CISIS), 2012 Sixth International Conference on* (pp. 12-19). IEEE

Dam, H. K., & Winikoff, M. (2013). Towards a next-generation AOSE methodology. *Science of Computer Programming*, *78*(6), 684-694.

Dam, K. H., & Winikoff, M. (2004). Comparing agent-oriented methodologies. In *Agent-Oriented Information Systems* (pp. 78-93). Springer Berlin Heidelberg.

De Oliveira, R., S. Gomes, H. Silva, A. Bittencourt and E.B. Costa, (2006). A Multi-agent based framework for supporting learning in adaptive automated negotiation. ICEIS, 4: 153-158.

Dechev, D., & Stroustrup, B. (2009). Reliable and Efficient Concurrent Synchronization for Embedded Real-Time Software. In *Space Mission Challenges for Information Technology, 2009. SMC-IT 2009. Third IEEE International Conference on* (pp. 323-330). IEEE.

DeLoach, S. A., Wood, M. F., & Sparkman, C. H. (2001). Multiagent systems engineering. *International Journal of Software Engineering and Knowledge Engineering*, *11*(03), 231-258.

Demir, O., & Lunze, J. (2012). Event-based synchronisation of multi-agent systems. IFAC Proceedings Volumes, 45(9), 1-6.

Drago, I., Mellia, M., M Munafo, M., Sperotto, A., Sadre, R., & Pras, A. (2012). Inside dropbox: understanding personal cloud storage services. In *Proceedings of the 2012 ACM conference on Internet measurement conference* (pp. 481-494). ACM.

Du, Y., Zhou, M., Childers, B., Melhem, R., & Mossé, D. (2013). Delta-compressed caching for overcoming the write bandwidth limitation of hybrid main memory. *ACM Transactions on Architecture and Code Optimization (TACO)*, *9*(4), 55.

Embi, M. A. (2011). E-Learning in Malaysia Higher Education Institutions: Status, *Trends & Challenges*. Department of Higher Education Ministry of Higher Education, Malaysia.

Evans, R., Kearney, P., Caire, G., Garijo, F., Gomez Sanz, J., Pavon, J., & Massonet, P. (2001). MESSAGE: Methodology for engineering systems of software agents. *EURESCOM, EDIN*, 0223-0907.

Ferragina, P., & Manzini, G. (2010). On compressing the textual web. In *Proceedings of the third ACM international conference on Web search and data mining* (pp. 391-400). ACM.

Ferragina, P., Nitto, I., & Venturini, R. (2011). On optimally partitioning a text to improve its compression. *Algorithmica*, *61*(1), 51-74.

Finin, T., Fritzson, R., McKay, D., & McEntire, R. (1994). KQML as an agent communication language. In *Proceedings of the third international conference on Information and knowledge management* (pp. 456-463). ACM.

France, R. B., Kim, D. K., Ghosh, S., & Song, E. (2004). A UML-based pattern specification technique. *Software Engineering, IEEE Transactions on*, *30*(3), 193-206.

Frankovič, B., & Oravec, V. (2005). Design of the agent-based intelligent control system. *Acta Polytechnica Hungarica*, *2*(2), 39-52.

Frias-Martinez, E., Chen, S. Y., & Liu, X. (2009). Evaluation of a personalized digital library based on cognitive styles: Adaptivity vs. adaptability.*International Journal of Information Management*, *29*(1), 48-56.

Friesen, N. (2005). Interoperability and learning objects: An overview of e-learning standardization. *Interdisciplinary Journal of E-Learning and Learning Objects*, *1*(1), 23-31.

Gallagher, P. S. (2007). *Assessing SCORM 2004 for its Affordances in Facilitating a Simulation as a Pedagogical Model*. Doctoral Dissertation, George Mason University, Fairfax, VA.

Ghobadi, A., Mahdizadeh, E. H., Kee, Y. L., Wei, L. K., & Ghods, M. H. (2011). Pre-processing directory structure for improved RSYNC transfer performance. In *Advanced Communication Technology (ICACT), 2011 13th International Conference on* (pp. 1043-1048). IEEE.

Giorgini P., & Henderson-Sellers B., (2005). Agent-Oriented Methodologies. Idea Group Publishing, USA.

Giorgini, P., & Henderson-Sellers, B. (2005). Agent-oriented methodologies: an introduction. *Agent-oriented Methodologies*, 1-19.

Gonçalves, G., Drago, I., da Silva, A. P. C., Vieira, A. B., & Almeida, J. M. (2014). Modeling the dropbox client behavior. In *2014 IEEE International Conference on Communications (ICC)* (pp. 1332-1337). IEEE.

Gonzalez-Barbone, V., & Anido-Rifon, L. (2008). Creating the first SCORM object. *Computers & Education*, *51*(4), 1634-1647.

Guarte, J. M., & Barrios, E. B. (2006). Estimation under purposive sampling. *Communications in Statistics—Simulation and Computation®*, *35*(2), 277-284.

Gunasekera, K., Zaslavsky, A., Krishnaswamy, S., & Loke, S. W. (2013). Building ubiquitous computing applications using the VERSAG adaptive agent framework. Journal of Systems and Software, 86(2), 501-519.

Gupta, D., & Sagar, K. (2010). RFSsingle-round algorithms. *International Journal of Computer Applications*, *4*(1), 32-36.

Hammami, S., & Mathkour, H. (2015). Adaptive e-learning system based on agents and object petri nets (AELS-A/OPN). *Computer Applications in Engineering Education*, 23(2), 170-190.

Hansen, C. K. (2010). Technology Trends in Mobile Communications How Mobile are Your Data?. *Invited paper for the IEEE Reliability Society Annual Technology Report. IEEE Transactions on Reliability*, *59*, 458-460.

Hegde, M. S., & Singh, S. (2013, August). Alert-BDI: BDI model with adaptive alertness through situational awareness. In *Advances in Computing, Communications and Informatics (ICACCI), 2013 International Conference on*(pp. 442-447). IEEE. https://en.wikipedia.org/wiki/GoDaddy

Huang, W., El-Darzi, E., & Jin, L. (2007). Extending the gaia methodology for the design and development of agent-based software systems. In *Computer Software and Applications Conference, 2007. COMPSAC 2007. 31st Annual International* (Vol. 2, pp. 159-168). IEEE.

Iglesias, C. A., Garijo, M., & González, J. C. (1999). A survey of agent-oriented methodologies. In *Intelligent Agents V: Agents Theories, Architectures, and Languages* (pp. 317-330). Springer Berlin Heidelberg.

Irmak, U., Mihaylov, S., & Suel, T. (2005). Improved single-round protocols for remote file synchronization. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE* (Vol. 3, pp. 1665-1676). IEEE.

Itmazi, J. A., Megías, M. G., Paderewski, P., & Vela, F. L. G. (2005). A comparison and evaluation of open source learning managment systems. In International Association for Development of the Information society (*IADIS) AC (Applied Computing)* (pp. 80-86).

Jansen, B. J., & Pooch, U. (2004). Assisting the searcher: utilizing software agents for Web search systems. *Internet Research*, *14*(1), 19-33.

Jennings, N. R. (1993). Specification and implementation of a belief-desire-joint-intention architecture for collaborative problem solving. *International Journal of Intelligent and Cooperative Information Systems*, *2*(03), 289-318.

Kammoun, H. M., Kallel, I., Casillas, J., Abraham, A., & Alimi, A. M. (2014). Adapt-Traf: An adaptive multiagent road traffic management system based on hybrid ant-hierarchical fuzzy model. Transportation Research Part C: Emerging Technologies, 42, 147-167.

Kanteev, M., Minakov, I., Rzevski, G., Skobelev, P., & Volman, S. (2007). Multi-agent meta-search engine based on domain ontology. In *Autonomous Intelligent Systems: Multi-Agents and Data Mining* (pp. 269-274). Springer Berlin Heidelberg.

Kazienko, P., & Kiewra, M. (2003). ROSA—Multi-agent system for web services personalization. In *Advances in Web Intelligence* (pp. 297-306). Springer Berlin Heidelberg.

Khan, Z. A. (2010). *Usability Evaluation of Web-based GIS Applications* (Doctoral Dissertation, Blekinge Institute of Technology).

Khemaja, M. (2014). Using Semantic Web Services technology for simulating collaborative learning activities: An approach based on ITSs' and e-learning standards extension. In *Computers in Education (SIIE), 2014 International Symposium on* (pp. 145-150). IEEE.

Kijewski-Correa, T., Kwon, D. K., Kareem, A., Bentz, A., Guo, Y., Bobby, S., & Abdelrazaq, A. (2012). SmartSync: An integrated real-time structural health monitoring and structural identification system for tall buildings. *Journal of Structural Engineering*, *139*(10), 1675-1687.

Kim, J., Yang, J., Shim, H., & Kim, J. S. (2013). Robustness of synchronization in multi-agent systems. In *Proc. 12th European Control Conf* (pp. 3821-3826).

Kotulski, L., Sędziwy, A., & Strug, B. (2015). Conditional Synchronization in Multi-agent Graph-based Knowledge System. *Procedia Computer Science*, *51*, 1043-1051.

Kundu, A., Guha, S. K., Chakraborty, T., Chakraborty, S., Pal, S., & Mukhopadhyay, D. (2009). Multi-Agent System for Search Engine based Web Server: A Conceptual Framework. *JDCTA: International Journal of Digital Content Technology and its Applications*, *3*(4), 44-59.

Kushilevitz, E., and Nisan, N., (2006) Computation time. Cambridge University Press, 2nd edition.

Labrou, Y., Finin, T., & Peng, Y. (1999). Agent communication languages: The current landscape. *IEEE Intelligent systems*, *14*(2), 45-52.

Lamport, L. (1986). The mutual exclusion problem: part I—a theory of interprocess communication. *Journal of the ACM (JACM)*, *33*(2), 313-326.

Leal, J. P., & Queirós, R. (2012). A comparative study on LMS interoperability. In Higher Education Institutions and Learning Management Systems: Adoption and Standardization (pp. 142-161). IGI Global.

Lee, Y., & Chong, Q. (2003). Multi-agent systems support for Community-Based Learning. *Interacting with computers*, *15*(1), 33-55.

Lee, Y., & Kozar, K. A. (2012). Understanding of website usability: Specifying and measuring constructs and their relationships. *Decision Support Systems*,*52*(2), 450-463.

Li, S., & Kokar, M. M. (2013). Agent Communication Language. In *Flexible Adaptation in Cognitive Radios* (pp. 37-44). Springer New York.

Li, Y., Xue, D., & Gu, P. (2008). Design for product adaptability. *Concurrent Engineering*, *16*(3), 221-232.

Lin, C. E., Kavi, K. M., Sheldon, F. T., Daley, K. M., & Abercrombie, R. K. (2007). A methodology to evaluate agent oriented software engineering techniques. In *System Sciences, 2007. HICSS 2007. 40th Annual Hawaii International Conference on* (pp. 60-60). IEEE.

Loseto, G., Scioscia, F., Ruta, M., & Di Sciascio, E. (2012). Semantic-based smart homes: a multi-agent approach. In *13th Workshop on Objects and Agents (WOA 2012)* (Vol. 892, pp. 49-55).

Luck, M., McBurney, P., & Preist, C. (2003). *Agent technology: Enabling next generation computing (a roadmap for agent based computing)*. AgentLink/University of Southampton.

Maalal, S., & Addou, M. (2012). A new approach of designing Multi-Agent Systems. *International Journal of Advanced Computer Science and Applications, Vol. 2, No. 11.*

Machado, M., & Tao, E. (2007). Blackboard vs. Moodle: Comparing user experience of learning management systems. In *Frontiers In Education Conference-Global Engineering: Knowledge Without Borders, Opportunities Without Passports, 2007. FIE'07. 37th Annual* (pp. S4J-7). IEEE.

Manzoor, U., & Ijaz, K. (2008). Optimizing bandwidth usage and supporting block level synchronization in MultiSync: a multiagent system for ubiquitous file synchronization Source. In *7th Proceedings of the WSEAS International Conference on Artificial intelligence, knowledge engineering and data bases, Cambridge, UK.*

Marikkannu, P., Jovin, J. A., & Purusothaman, T. (2011). Fault-tolerant adaptive mobile agent system using dynamic role based access control.*International Journal of Computer Applications, 20*(2), 1-6.

Marsh, D., Tynan, R., O'Kane, D., & O'Hare, G. M. (2004). Autonomic wireless sensor networks. Engineering Applications of Artificial Intelligence, 17(7), 741-748.

Martínez, J. Á., & Navarra, P. L. (2007). Content interoperability on e-learning platforms: standardization, digital libraries, and knowledge management. *RUSC. Universities and Knowledge Society Journal*, *3*(2).

McIntosh, D. (2014). Vendors of learning management and e-learning products. *Learning Manage. Vendors*, 88-96.

McKay, D., Finin, T., Fritzson, R., & McEntire, R. (1994). *KQML: a language and protocol for knowledge and information exchange*. Technical report CS-94-02, Computer Science Department, University of Maryland and Valley Forge Engineering Center, Unisys Corporation.

Meenakumari, D. J., & Antony, B. (2013). Assessing the Essential Features of ICT-Based LMS for Performance Enhancement. *IRACST-International Journal of Research in Management & Technology (IJRMT), ISSN*, 2249-9563.

Meisam Hejazinia, Mohammad Reza Razzazi, (2010). M-Learning System over MANET on Mobile phones, 4th International Conference on Distance Learning and Education (ICDLE) IEEE.

Miranda, J. H., Sanchez, D. V., Ceballos, D. E. V., Rubio, E. H., Viveros, A. M., & Ledezma, E. F. R. (2014). Middleware to integrate Heterogeneous Learning Management Systems and initial results. *Editorial Preface*, *5*(10).

Mohd Khazani Abdullah, Nur Riza Mohd Suradi, Nordin Jamaluddin, Ahmad Samsuri Mokhtar , A. Rahim Abu Talib, Mohd Faizal Zainuddin, (2006), K-CHART: A TOOL FOR RESEARCH PLANNING AND MONITORING, *Journal of Quality Measurement and Analysis*, JQMA 2(1), 123-129.

Nedhal A. Al-Saiyd, Muna F. Al-Samarae, Intisar A. Al-Sayed,, (2014), A Multi-Agent Systems Engineering For Semantic Search Of Reuse Software Components, *International Journal of Computer Science*, Volume 11, Issue 6, 81-88.

Nguyen, P. T., Schau, V., & Rossak, W. R. (2011). Towards an adaptive communication model for mobile agents in highly dynamic networks based on swarming behaviour. In The 9th European Workshop on Multi-agent Systems (EUMAS).

Niazi, M., Manzoor, U., Summiya, K. I., & Saleem, H. (2005). Eliminating duplication and ensuring file integrity in Multisync: A multiagent system for ubiquitous file synchronization. In *9th International Multitopic Conference, IEEE INMIC 2005* (pp. 1-6). IEEE.

Nielsen, J. (1994). Heuristic Evaluations. In: J. Nielsen & R.L. Mack. (Eds), *Usability Inspection Methods*. New York: John Wiley & Sons.

Noel, P., Suel, T., & Trendafilov, D. (2004). Improved file synchronization techniques for maintaining large replicated collections over slow networks. In *Data Engineering, 2004. Proceedings. 20th International Conference on* (pp. 153-164). IEEE.

Nor, M.Z.M., R. Abdullah, M.H. Selamat and M.A.A. Murad, (2009). Knowledge sharing interactions in collaborative software maintenance environment. *Proceedings of the International Conference Computer Technology and Development*, Nov. 13- 15, IEEE Xplore Press, pp: 201-205.

Norling, E. J. (2007). PhD Thesis, *Modelling Human Behaviour with BDI Agents*. University of Melbourne, Department of Computer Science and Software Engineering.

Ogunnusi, O.S. and S.A.D. Razak, (2013). Research on security protocol for collaborating mobile agents in network intrusion detection systems. *Am. J. Applied Sci*., 10: 1638-1647.

Orlitsky, A., & Viswanathan, K. (2001). Practical protocols for interactive communication. In *Information Theory, 2001. Proceedings. 2001 IEEE International Symposium on* (p. 115).

P.N.V.Pavan, Prof. H. Santhi, Prof. N. Jaisankar, (2012). A Survey on M-Learning, *International Journal of Computer Applications (0975 – 888) Volume 48– No.3.*

Padgham, L., & Winikoff, M. (2002). Prometheus: A pragmatic methodology for engineering intelligent agents. In *Proceedings of the OOPSLA 2002 Workshop on Agent-Oriented Methodologies* (pp. 97-108).

Padgham, L., Thangarajah, J., & Winikoff, M. (2007). AUML protocols and code generation in the Prometheus design tool. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*(p. 270). ACM.

Padgham, L., Thangarajah, J., & Winikoff, M. (2008). The prometheus design tool–a conference management system case study. In *Agent-Oriented Software Engineering VIII* (pp. 197-211). Springer Berlin Heidelberg.

Padovitz, A., Zaslavsky, A., Loke, S. W., & Tosic, M. (2004). Agent communication using publish-subscribe genre: architecture, mobility, scalability and applications. *Ann. Math. Comput. Teleinform.(AMCT)*, *1*(2), 35-50.

Pappas, C., (2013) available at: http://elearningindustry.com/learning-management-systems-comparison-checklist-of-features. Retrieved on March 2015.

Passadore, A., Grosso, A., & Boccalatte, A. (2009). An agent-based semantic search engine for scalable enterprise applications. *Proceedings of ONTOSE*, 83.

Peng, L.K., Ramaiach, C.K, & Foo, S. (2004). Heuristic-Based User Interface Evaluation at Nanyang Technological University in Singapore. *Program: Electronic Library and Information Systems*, 38 (1): 42-59.

Perez, A.B.G. and J.A.R. Uresti, (2014). Strategy patterns prediction model. *J. Comput. Sci*., 10: 73-84. DOI: 10.3844/jcssp.2014.73.84.

Peter, B., & Shane, G. (2013). Choosing a Learning Management System.*Advanced Distributed Learning (ADL) Co-Laboratories Report*

Pierre, S., Hoang, H. H., & Pelletier, S. J. (2003). Modeling a multi-agent system for retrieving information from distributed sources. *CIT. Journal of computing and information technology*, *11*(1), 15-39.

Qian, H., Yafeng, G., Yong, W., & Baohua, Q. (2013). A web site protection oriented remote backup and recovery method. In *Communications and Networking in China (CHINACOM), 2013 8th International ICST Conference on* (pp. 395-399). IEEE.

Queirós, R., & Leal, J. P. (2013). A survey on eLearning content standardization. In *Information Systems, E-learning, and Knowledge Management Research* (pp. 433-438). Springer Berlin Heidelberg.

Rasch, D., & Burns, R. C. (2003). In-Place Rsync: File Synchronization for Mobile and Wireless Devices. In *USENIX Annual Technical Conference, FREENIX Track* (pp. 91-100).

Ricci, A., Piunti, M., & Viroli, M. (2011). Environment programming in multi-agent systems: an artifact-based perspective. *Autonomous Agents and Multi-Agent Systems*, *23*(2), 158-192.

Riddell, R., (2013) available at: http://www.educationdive.com/news/12-learning-management-system-providers-and-what-they-bring-to-classrooms/97613/. Retrieved on March 2015.

Rodríguez, S., De Paz, J. F., Villarrubia, G., Zato, C., Bajo, J., & Corchado, J. M. (2015). Multi-agent information fusion system to manage data from a WSN in a residential home. Information Fusion, 23, 43-57.

Rubin, B., Fernandes, R., Avgerinou, M. D., & Moore, J. (2010). The effect of learning management systems on student and faculty outcomes. *The Internet and Higher Education*, *13*(1), 82-83.

Rumbaugh, J., Jacobson, I., & Booch, G. (2004). *Unified Modeling Language Reference Manual, The*. Pearson Higher Education.

Ryann K. Ellis (2009), A field guide to Learning Management Systems, American Society for Training & Development (ASTD). ASTD Inc

Sajja, P.S., (2008). Multi-agent system for knowledge based access to distributed databases. Interdisciplinary J. Inform. Knowl. Manage., 3: 1-9.

Sakai project website available at: https://sakaiproject.org/. Retrieved March 2015.

Sala, F., Schoeny, C., Bitouzé, N., & Dolecek, L. (2016). Synchronizing files from a large number of insertions and deletions. *IEEE Transactions on Communications*, *64*(6), 2258-2273.

Santos, L., (2009). Adaptability Support in a Learning Management System. Master Thesis, Universidade de Lisboa, Lisbon.

Schoeny, C., Bitouzé, N., Sala, F., & Dolecek, L. (2014). Efficient file synchronization: Extensions and simulations. In *Signals, Systems and Computers, 2014 48th Asilomar Conference on* (pp. 2129-2133). IEEE.

SCORM 2004 4th Ed. Testing Requirements (TR) Version 1.1 (2009) Advanced Distributed Learning. Available at: http://www.adlnet.gov/wp-content/uploads/2011/07/SCORM_2004_4ED_v1_1_TR_20090814.pdf. Retrieved on June 2014.

SCORM 2004 Handbook version 1.04 (2006). The e-Learning Consortium, Japan. Available at http://203.183.1.152/aen/content/act2005eg/data/txt1.pdf. Retrived on March 2015

Shaoyan, C., Xianwei, H., & Ming, L. (2009). Research of Mobile Forensic Software System Based on Windows Mobile. In *Wireless Networks and Information Systems, 2009. WNIS'09. International Conference on* (pp. 366-369). IEEE.

Sheng, Y., Xu, D., & Wang, D. (2010). A two-phase differential synchronization algorithm for remote files. In *Algorithms and Architectures for Parallel Processing* (pp. 65-78). Springer Berlin Heidelberg.

Shoham, Y. (1993). Agent-oriented programming. *Artificial intelligence*, *60*(1), 51-92.

Silva, D. C., Braga, R. A., Reis, L. P., & Oliveira, E. (2012). Designing a meta-model for a generic robotic agent system using Gaia methodology. *Information Sciences*, *195*, 190-210.

Sirott, J., Sun, L. C., & Denbo, D. W. (2005). Networked synchronization of netCDF datasets. In *21st International Conference on Interactive Information Processing Systems*.

Sloane, A., & Lawrence, D. (Eds.). (2012). *Multimedia Internet broadcasting: quality, technology and interface*. Springer Science & Business Media.

Ssemugabi, S., & De Villiers, R. (2007). A comparative study of two usability evaluation methods using a web-based e-learning application. In *Proceedings of the 2007 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries* (pp. 132-142). ACM

Ssemugabi, S., & De Villiers, R. (2007). A comparative study of two usability evaluation methods using a web-based e-learning application. In *Proceedings of the 2007 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries* (pp. 132-142). ACM.

Sterling, L. (1994). *The art of Prolog: advanced programming techniques*. MIT press.

Su, C. J., & Wu, C. Y. (2011). JADE implemented mobile multi-agent based, distributed information platform for pervasive health care monitoring. *Applied Soft Computing*, *11*(1), 315-325.

Suel, T., and Memon, N. (2002). Lossless Compression Handbook, chapter ”Algorithms for delta compression and remote file synchronization”, Academic Press, 2002.

Sun, H., Thangarajah, J., & Padgham, L. (2010). Eclipse-based prometheus design tool. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1* (pp. 1769-1770). International Foundation for Autonomous Agents and Multiagent Systems.

Talib, A.M., R.B. Atan, R. Abdullah and M.A.A. Murad, (2012). Towards a comprehensive security framework of cloud data storage based on multi agent system architecture. *J. Inform. Security*, 3: 295-306.

Teodosiu, D., Bjorner, N., Gurevich, Y., Manasse, M., & Porkka, J. (2006). Optimizing file replication over limited bandwidth networks using remote differential compression. *Microsoft Research TR-2006-157*.

The Eclipse Platform, http://www.eclipse.org/

Thompson, S. K. (2012). Simple random sampling. *Sampling, Third Edition*, 9-37.

Tridgell A. (2000). *Efficient Algorithms for Sorting and Synchronization*. PhD thesis, Australian National University.

Tridgell A. and P. MacKerras. P. (1996). The rsync algorithm. Technical Report TR-CS-96-05, Australian National University.

Trotter, A. (2008). Blackboard vs. Moodle: Competition in course-management market grows. *Education Week’s Digital Directions: Trends and Advice for K-12 Technology Leaders, 2*, 21.

Tynan, R., Marsh, D., O'kane, D., & O'Hare, G. M. (2005). Agents for wireless sensor network power management. In *Parallel Processing, 2005. ICPP 2005 Workshops. International Conference Workshops on* (pp. 413-418). IEEE.

Unal, Z., & Unal, A. (2011). Evaluating and comparing the usability of web-based course management systems. *Journal of Information Technology Education: Research*, *10*(1), 19-38.

Wahlbeck, K., Cheine, M., Essali, A., & Adams, C. (2014). Evidence of clozapine’s effectiveness in schizophrenia: a systematic review and meta-analysis of randomized trials. *American Journal of Psychiatry*.

Walczak, A., Braubach, L., Pokahr, A., & Lamersdorf, W. (2007). Augmenting BDI agents with deliberative planning techniques. In *Programming Multi-Agent Systems* (pp. 113-127). Springer Berlin Heidelberg.

Wang, J., & Liu, C. (2006). Agent-Oriented probabilistic logic programming with fuzzy constraints. In *Agent Computing and Multi-Agent Systems* (pp. 664-671). Springer Berlin Heidelberg.

Wang, J., & Senecal, S. (2007). Measuring perceived website usability. *Journal of Internet Commerce*, *6*(4), 97-112.

Wang, J., & Senecal, S. (2007). Measuring perceived website usability. *Journal of Internet Commerce*, *6*(4), 97-112.

Wang, K. H. C., & Botea, A. (2008). Fast and Memory-Efficient Multi-Agent Pathfinding. In *ICAPS* (pp. 380-387).

Weyns, D., Steegmans, E., & Holvoet, T. (2004). Protocol based communication for situated multi-agent systems. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems-Volume 1* (pp. 118-125). IEEE Computer Society.

Wong, E. M. L., Chan, S. W. C., & Chair, S. Y. (2010). Effectiveness of an educational intervention on levels of pain, anxiety and self-efficacy for patients with musculoskeletal trauma. *Journal of Advanced Nursing*, *66*(5), 1120-1131.

Wood, M. F., & DeLoach, S. A. (2001). An overview of the multiagent systems engineering methodology. In *Agent-Oriented Software Engineering* (pp. 207-221). Springer Berlin Heidelberg.

Wooldridge M (2002). An Introduction to MultiAgent Systems. John Wiley & Sons, Chichester, UK, http://www.csc.liv.ac.uk/~mjw/pubs/imas/.

Wooldridge, M. (1998). Verifiable semantics for agent communication languages. In *Multi Agent Systems, 1998. Proceedings. International Conference on* (pp. 349-356). IEEE.

Wooldridge, M. and N.R. Jennings, (1995). Intelligent agents: Theory and practice. Knowl. Eng. Rev., 10: 115-152.

Wooldridge, M., & Jennings, N. R. (1995). Agent theories, architectures, and languages: a survey. In *Intelligent agents* (pp. 1-39). Springer Berlin Heidelberg.

Wooldridge, M., Jennings, N. R., & Kinny, D. (2000). The Gaia methodology for agent-oriented analysis and design. *Autonomous Agents and Multi-Agent Systems*, *3*(3), 285-312.

Wooldridge, M., Jennings, N. R., & Kinny, D. (2000). The Gaia methodology for agent-oriented analysis and design. *Autonomous Agents and multi-agent systems*, *3*(3), 285-312.

Xiaohu, W., Lele, W., & Nianfeng, L. (2012). A Fast Search Algorithm Based on Agent Association Rules. *Physics Procedia*, *25*, 1022-1026.

Xu D., Liu C., and Wu J., (2011) “Remote File Synchronization: A Performance Comparison and Analysis of Different Approaches”. The 25th International Conference on Advanced Information Networking and Applications Symposium on Frontiers of Information Systems and Network Applications. Singapore.

Yan, H., Irmak, U., & Suel, T. (2008). Algorithms for low-latency remote file synchronization. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*.

Yang, W., Gu, C., Wan, F., Hanyu, S., & Jiping, Z. (2014). A cloud storage synchronization strategy based on cdmi. In *Innovative Computing Technology (INTECH), 2014 Fourth International Conference on* (pp. 120-129). IEEE.

Zambonelli, F., Jennings, N. R., & Wooldridge, M. (2003). Developing multiagent systems: The Gaia methodology. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, *12*(3), 317-370.

Zhang, S., He, Z., & Yang, H. (2012). Mobile agent routing algorithm in wireless sensor networks. In *Advances in Computer Science and Information Engineering* (pp. 105-113). Springer Berlin Heidelberg.

Zhou Xiang, (2008). Design and implementing an integrated Learning Content Management System**,** Workshop on Knowledge Discovery and Data Mining IEEE.