ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 11, November 2016

Comparative Study of Agent Oriented Methodologies in e-Learning systems

Gopal Sakarkar[1], S. P. Deshpande[2], V. M. Thakare[3]

Assistant Professor, Dept. of Computer Applications, G H Raiosni College of Engineering, Nagpur, India[1]

Head of Department, P.G. Dept. of Computer Science and Technology,H.V.P. Mandal, Amravati, India[2]

Professor and Head of Department, S.G.B. Amravati University, Amravati, India[3]

ABSTRACT: e-Learning is a new paradigm offered by developed and developing countries for delivering effective learning materials. Providing a personalize web based e-Learning system is an challenging research area for today . Software Agents are somewhat helpful to provide the personalize data using some intelligent technique .To develop such agent oriented system , its required detail and systematic analysis of that particular domain. This paper provided study the various important agent oriented methodologies and compare them on various criteria like number of phases required for implementation, mobility support , framework /tool used for development and usefulness in multi agent system.

KEYWORDS: e-Learning, Software Agents, Methodology, Semantic Web.

I. INTRODUCTION

In recent years, the concept of web-based intelligent learning and the use of Internet in teaching and learning has been received increasing attention. People use the Internet and new technologies every day for information sharing , communication, entertainment, purchasing goods and services, and reading learning materials. With this increasing availability of the Internet, it is very challenging task that, what and how to deliver the learning materials to students across the world, which will be improve evolution of e-Learning.

One of the solution to overcome problem of static information provider of e-learning is by using a personalization agent in an online e-learning system that retrieve learning materials based on cognitive style, personal preferences and prior knowledge[1].

In developing an Intelligent Learning Management Systems, computer technologies and Artificial Intelligence are used to guide and provide e-learning. An agent recommends activities to a learner based on his access history [2].

To improve the efficiency of education , the internet technology used to design, implement , manage, support and extend learning methodologies .e-Learning provided flexibility, diversity , measurement and online storage with cloud computing model [3].

In developing nation like India it is very vital to implement the E-Learning software solution to improve their educational standards. Due to improper facilities and the lack of infrastructure there lie many problems in their part of implementation in much educational institution. The cloud with E-Learning is the last alternate to this lack and it fulfills the need[4].

E-learning has become one of the primary alternatives for distributing education around the globe. Many low-resource countries suffer from a shortage of teachers and ICT can improve the access to education and improve the ability for marginalized groups to attend school[5].

E-learning provide the best interaction between student and teacher at electronically, this phenomena based on chatting system or another media for virtual platform we have many software platform available Good like Moodle and Blackboard learning system[6].

DOI: 10.15680/IJIRCCE.2016. 0411172

II. INTELLIGENCE E-LEARNING SYSTEMS

Agent based e-Learning can manage the information overload, serve as academic experts, and create programming environments for the learners. There are several characteristics specific to e-Learning system such as Interaction, Personalization, Adaptation, Intelligence, Interoperability, Accessibility and Security. Agent based architecture for e-Learning system provides the above mentioned features to support instructional design, to retrieve relevant learning materials, to process and analyze data to enable meaningful e-Learning recommendations to instructors and learners [7]. Artificial intelligence plays an important role in providing more effective, advanced e-learning systems. Artificial intelligence based evaluation mechanism is able to determine which material will be viewed next according to students' learning levels[8].

An intelligent prediction model used for learners outcome forecasting approach, which helps facilitators and users discover more interesting knowledge information and predict the learning outcomes. It is useful to overcome traditional machine learning techniques that includes objective prediction and subjective forecasting methods [9].

The proposed [10] Intelligent Tutor System for JAVA that has been improved the student learning productivity by 10%. In which the system automatically produces exercises and during the process of exercising it uses feedback information from the student performance, it has a capability of evaluating and monitoring of student learning process.

The proposed system consists of ontology for the e-learning process, such as teaching methods, learning styles, learning activities and course syllabus , which is capable of gaining user adaptability, performance scalability and concept reusability. It also has ability to act in an intelligent manner by evaluating the academics initially and then provide personalized suggestions to the academics indicating their weaknesses and strengths[11].

III. SOFTWARE AGENTS

Software agents are software entities that can act on behalf of user in an autonomous fashion, and perform their action in some level of proactively and reactively [12].

It is a persistent software entity dedicated to a specific purpose and also it is computer program that simulate a human relationship by doing something that another person could do for you [13].

It is software entity, which functions continuously and autonomously in a particular environment, often inhabited by other agents and process [14].

With this, it is better to fix up the definition as given below by considering the fact that Autonomous agents are not directed by commands from a user. Instead, they are directed by a set of tendencies [15].

3.1 Characteristics of Software Agent

Autonomy: Software agent should be able to perform the majority of their problem solving without the direct intervention of human or other agents and they should have a degree of control over own action and their own internal[16].

Social ability: Software agent should be able to interact when they deem appropriate with other software agents and human in order to complete own problem solving and to help other with their activities where appropriate[16].

Mobility: Its means that the agent is able to transport itself from one machine to another and across different system architectures and platforms [17].

Intelligence: The agent's state is formalized by knowledge and interacts with other agents using symbolic language [17].

3.2 Agent Architecture Type

1) Logic based agents: In which decision making is realized through logical inference.
2) Reactive agents: In which decision-making is implemented in some form of direct mapping from situation to action .
3) Belief-Desire-Intention (BDI) agents: In which decision-making depends upon the manipulation of data structures representing the beliefs, desires, and intentions of the agent.
4) Layered architectures: In which decision making is realized via various software layers, each of which is more-or-less explicitly reasoning about the environment at different levels of abstraction.

IV. AGENT ORIENTED METHODOLOGIES

Agent-oriented methodologies have several roots. Some are based on ideas from artificial intelligence (AI), others as direct extensions of existing OO methodologies, whilst yet others try to merge the two approaches by taking a more purist approach yet allowing OO ideas when these seem to be sufficient[18].

Agents have a more complex behavior and structure than objects, and in this respect they are more comparable to subsystems in some OO methodologies.. Their internal structure differs from objects in that agents have a more complex underlying functional architecture such as the belief-desire-intention (BDI) architecture . In this respect they are on a higher level of abstraction than objects [19.].

Agents' plans can be directly implemented using an agent programming language or an appropriate tool or environment such as DASEDIS [20]. Specifically using such tools, implementing an agent is mainly specifying its plans.

4.1 Important and popular Agent Oriented Methodologies are

- GAIA
- .MaSE
- MESSAGE
- MaMM
- MGAIA
- Tropos
- MAS CommonKADS
- Prometeus

- Generic Architecture for Information Availability

Gaia is a methodology that has been highly influenced by object technology yet retains a truly agent-oriented feel. The Gaia methodology is both general, in that it is applicable to a wide range of multi-agent systems, and comprehensive, in that it deals with both the macro-level (societal) and the micro-level (agent) aspects of systems. Gaia is founded on the view of a multi-agent system as a computational organisation consisting of various interacting roles [4].

The first proposed Gaia methodology consists of two iterative phases, analysis and design. Gaia does not address requirement elicitation but does not necessarily ignore it. Gaia is usually open to any platform of implementation but experience shows that it is better implemented with a FIPA compliant platform like JADE [5].

- Multiagent Systems Engineering

The Multiagent Systems Engineering (MaSE) were introduced by Wood and Loach [6]is a fully describes the process which guides a system developer from an initial system specification to system implementation. It provide a complete-lifecycle methodology to assist system developers to design and develop a multi-agent system[7].

MaSE methodology is basically divided into four components that use in the distributed agent paradigm [8].

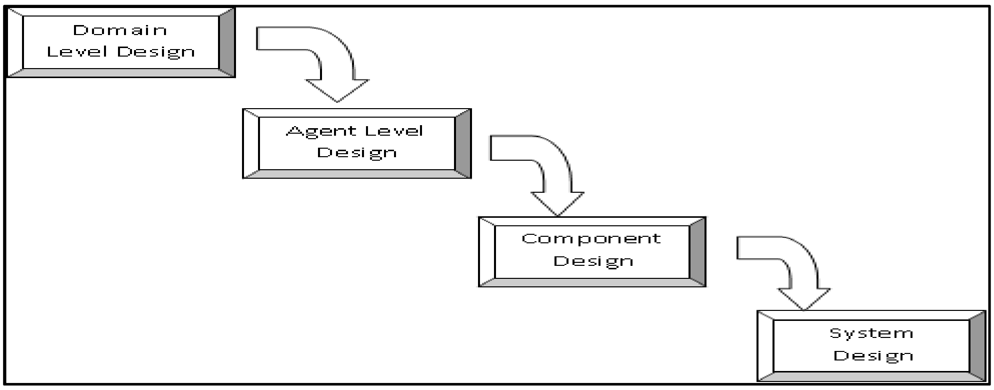

Fig 1 . Workflow of MaSE Methodology

MaSE methodology is basically divided into four components that use in the distributed agent paradigm [24].
1. Domain Level Design
The first step in MaSE is domain level design, which captures the basic types and interactions between agents in our system.
2. Agent Level Design
In this level , define (or reuse) the agent architectures for each individual agent type.
3. Component Design
The component design level is the next obvious level of the MaSE methodology.
4 .System Design

Finally, system design takes place once the design of the domain, agents, and components are complete.

- Methodology for Engineering Systems of Software Agents

The MESSAGE (Methodology for Engineering Systems of Software Agents) methodology extends the UML meta-model adding new meta-concepts (such as Agent, Goal and Task) in order to provide developers with a higher degree of expressiveness. A proper notation is also defined to represent graphically instances of these meta-concepts. Furthermore a process and a number of heuristic rules are provided to guide the developer in carrying out the analysis and design of a system under development[25] .

MESSAGE is also an AOSE methodology which builds upon current software engineering best practices covering analysis and design of MAS which is appropriate for use in mainstream software engineering departments. It has well defined concepts and a notation that is based on UML whenever appropriate [26].

- Mobile agent Mobility Methodology

Mobile agent Mobility Methodology is the methodology for modeling mobility in mobile agent-based systems using underlying principles of Genetic Algorithms (GA) with emphasis on fitness functions and genetic representation [27].

- mGAIA

It is extending the GAIA methodology to model mobile agent systems , is basically used during the development of several mobile agent implementation environments that has necessitated conceptual modeling techniques for mobile agent applications[28].
It does not handle mobility in the analysis phase but instead it models it in the design phase. one of limitation of it , that mGAIA does not fully develop the concept of agent’s visit. [29]

- Tropos

Tropos is based on two key ideas. First, the notion of agent and all related mentalistic notions (for instance goals and plans) are used in all phases of software development,from early analysis down to the actual implementation. Second, Tropos covers the very early phases of requirements analysis, thus allowing for a deeper understanding of the environment where the software must operate, and of the kind of interactions that should occur between software and human agents [30].

- Multi Agent System-CommonKADS [31] is a methodology designed for the development of knowledge based systems (KBS) analogous to methods of software engineering. methodology follows an approach to Knowledge Based System Development as the building of a number of separate models that capture salient features of the system and its environment.

Normally MAS-CommonKADS methodology has six phases for multi agent development system namely

I. Conceptualization
II. Analysis
III. Design
IV. Coding and testing of each agent

V. Integration
VI. Operation and maintenance

- Prometheus

The Prometheus methodology is used for developing agent systems , as it has an iterative approach that should be applied, but it has no specific mechanisms to support such. Prometheus consists of three main design phases, plus implementation and testing .The design phases are: (i) System Specification, (ii) Architectural Design, (iii) Detailed Design [32].

Prometheus is supported by two tools. The JACK Development Environment (JDE), developed by Agent Oriented Software (www.agent-software.com) includes a design tool that allows overview diagrams to be drawn. The Prometheus Design Tool (PDT) provides forms to enter design entities. It performs cross checking to help ensure consistency and generates a design document along with overview diagrams . Its aim to the development of intelligent agents in particular BDI agents i.e. Uses goals, beliefs, plans, and events, it mainly used in industrial software development, not researchers [33].

V. PROPOSED IMSA METHODOLOGY

The proposed Intelligent Mobile Software Agent methodology.

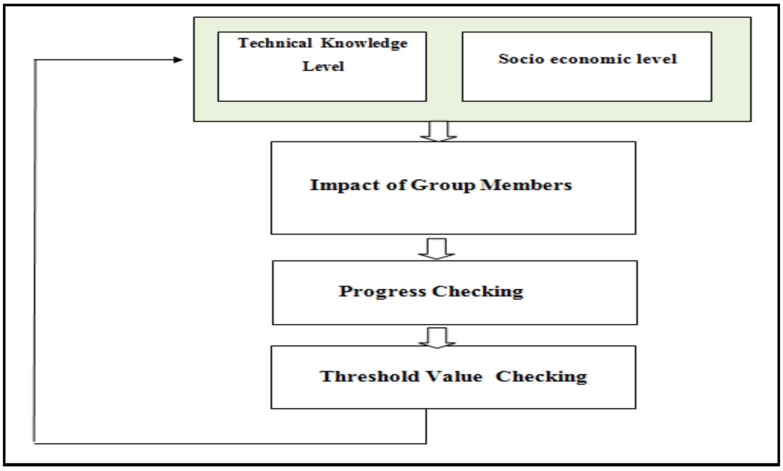

Fig 2 . Working of IMSA Methodology

This proposed methodology is a continuous process of improving the user's knowledge up to the threshold mark.

In this methodology, first stage is to check the users technical knowledge by taking a basic C language technical test, after that (s)he has to select the minimum three technical skills like C,C++,JAVA, Advance JAVA, HTML, etc. Also the system has to classify the user according to its socioeconomic status level by considering his personal information, parental information and educational information.

This first classification is totally depending on technical knowledge and socio economic status of users.

At the second stage, proposed methodology has utilized the impact of neighbourhood size that plays an important role to provide accurate recommendation and information to learners. Using the socio-economic level of a leaner, system can fairly differentiate learners in to appropriate group.

Third stage is used to continuously check the progress of learner on the basis of his utilization of e-Learning pedagogy. The feedback is given to recommend web links, time duration spend to learn the topic and overall utilization of learning system.

At particular threshold level (as in this work , it is 75%) of progress , the leaner allowed to re-appear for C language technical test and improved his level from current level to expert level.

The proposed work used Level 3 Software Agent Intelligence, which is top most level of intelligence. As per this level, Agent has learning and deductive components to help the user who cannot formalize a query or specify a target for a search.

VI. COMPARATIVE STUDY OF AGENT ORIENTED (AO) METHODOLOGIES

To find out the best methodology among the number of agent oriented methodologies is very challenging task, proposed the comparison of a various AO , based on number of phases required for implementation, supporting for mobility , framework /tool used for development and usefulness in multi agent system.

Table 1. Comparison of Agent Oriented Methodologies

Sr.No.	Methodology Name	Number of Phases	Support for Mobility	Developing Framework	Multi Agent System Support
1	GAIA	2	N	JADE	Y
2	MaSE	4	N	AgentTool	N
3	MESSAGE	2	N	MetaEdit+	Y
4	MaMM	3	Y	Matlab	N
5	mGAIA	2	Y	GrassHooper	Y
6	TROPOS	5	N	T-Tool	Y
7	MAS CommonKADS	6	N	JavaCase Tool and Rational Rose	Y
8	PROMETHEUS	3	N	JACK	Y
9	IMSA	4	Y	JAVA	N

VII. CONCLUSION

Methodology is a process of methods used in domain study and its implementation .This paper cover the various Agent Oriented methodologies that used to develop the agent based systems and analysis them according various common criteria.

Choosing the proper methodology is depending on the requirement of system application .

REFERENCES

1. Sivakumar, K.Vivekanandan,B.Arthi,S.Sandhya,Veenas Katta , Incorporating Agent Technology for Enhancing the Effectiveness of e-learning System ,IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 3 ,2011,pp.454-460
2. S.Bhaskaran and P. Swaminathan , Intelligent Adaptive E-learning Model for Learning Management System , Research Journal of Applied Sciences, Engineering and Technology Vol.7, Issue 16, 2014 ISSN: 2040-7459
3. Ghazal Riahi , E-Learning Systems Based on Cloud Computing: A Review, International Conference on Soft Computing and Software Engineering, ELSEVIER Procedia Computer Science 62, 2015, pp-352-359
4. Dr. N.Vanjulavalli,S.Arumugam, Dr.A.Kovalan ,An Effective tool for Cloud based E-learning Architecture, International Journal of Computer Science and Information Technologies, Vol. 6 (4) , 2015,pp- 3922-3924
5. Mohammed Omer, Tina Klomsri, Matti Tedre, Iskra Popova, Marie Klingberg-Allvin, Fatumo Osman, E-learning Opens Door to the Global Community: Novice Users' Experiences of E-learning in a Somali University. MERLOT Journal of Online Learning and Teaching Vol. 11, No. 2, June 2015,pp-267-279
6. Kanak Sachan,Dr. Rajiv Singh,A SURVEY AND COMPARETIVE ANALYSIS OF E-LEARNING PLATFORM (MOODLE AND BLACKBOARD), International Journal of Recent Research in Mathematics Computer Science and Information Technology Vol. 2, Issue 1, September 2015, pp: 293-299
7. K. Sakthiyavathi,K. Palanivel A Generic Architecture for Agent Based E-Learning System, International Conference on Intelligent Agent & Multi-Agent Systems, July 2009, pp.1-5.
8. Utku KOSE, Dr. Ahmet ARSLAN , E-LEARNING EXPERIENCE WITH ARTIFICIAL INTELLIGENCE SUPPORTED SOFTWARE: An International Application on English Language Courses , GLOKALde July 2015, ISSN 2148-7278, Volume: 1 Number: 3 Article: 3 ,pp-61-75
9. M. Ravichandran,G. Kulanthaivel , Intelligent prediction model for learners outcome forecasting in e-learning,IEEE International Conference on Computing and Communications Technologies (ICCCT), 2015 , pp-7 – 11
10. NEBOJŠA GAVRILOVIĆ, SLOBODAN JOVANOVIĆ ,IMPLEMENTING ITS (INTELLIGENT TUTORING SYSTEMS) FOR JAVAPROGRAMMING E-LEARNING, The Sixth International Conference on e-Learning (eLearning-2015), 24- 25 September 2015, Belgrade, Serbia
11. Pankajdeep Kaur,Pallavi Sharma and Nikhil Vohra,An Ontology Based E-Learning System, International Journal of Grid Distribution Computing Vol. 8, No.5, (2015), pp.273-278
12. Q. H. Mahmoud and L. Yu,An Architecture and Business Model for Making Software Agents Commercially Viable', 38th Hawaii Int. Conf. on System Sciences-2005,IEEE

13. N. Jennings and M. Woodldridge ,Software Agent, IEEE Review ,Jan. 1996.pp17-20
14. K. Decker,K. Sycara and M. Williamson, Intelligent Adaptive Information Agents
15. Hemme-Unger K., Flor T., Vögler G.MDA Development Approach for Pervasive Computing, OOPSLA'03, Anaheim, CA, USA, 18th ACM SIGPLAN conf. on OOP, systems, languages, and app., Oct. 26-30, 2003
16. Chira, C., January 31, 2007, *Software agents.* (IDIMS Report)
17. Etzioni, O., & Weld, D. S., Intelligent agents on the Internet: Fact, fiction and forecast, *IEEE Expert, 10*(4), 1995, pp. 44-49
18. Paolo Giorgini , Brian Henderson-Sellers Agent-Oriented Methodologies: An Introduction ,
19 A. S. Rao, M. P. Georgeff: An Abstract Architecture for Rational Agents in B.Nebel, C. Rich, W. Swartout (eds.): Proc. International Conference on Principles of Knowledge Representation and Reasoning (KR-92), Morgan Kaufmann, San Mateo, 1992
20. B. Burmeister: DASEDIS – Eine Entwicklungsumgebung zum Agenten-Orientierten Programmieren in: H.J. Müller (ed.): Verteilte Künstliche Intelligenz, BI– Wissenschaftsverlag, Mannheim, 1993.
21. Wooldridge, M., Jennings, N.R., & Kinny, D. (2000). The Gaia methodology for agent-oriented analysis and design. Journal Autonomous Agents and Multi-Agent Systems, 3, 285-312
22. A. Tveit, 2001,A survey of Agent-Oriented Software Engineering, Norwegian University of Science and Technology, May 8.
23. Khanh Hoa Dam , Michael Winikoff Comparing AgentOriented Methodologies , ACM 089791886/ 97/05
24. Scott A. DeLoach , Multiagent Systems Engineering: A Methodology And Language for Designing Agent Systems, International Conference on Agent-Oriented Information Systems (AOIS) '99,pp-1-9
25. Giovanni Caire ,Francisco Leal , MESSAGE: Methodology for Engineering Systems of Software Agents , EURESCOM Technical Information, September 2001 ,pp-1-27
26. Giovanni Caire,Wim Coulier,Francisco Garijo,Jorge Gomez,Juan Pavon, Francisco Leal, Paulo Chainho, Paul Kearney, Jamie Stark, Richard Evans, Philippe Massonet ,Agent Oriented Analysis using MESSAGE/UML.
27. Melomey, Divina A (2012) *A Methodology for Modelling Mobile Agent-Based Systems (Mobile agent Mobility Methodology – MaMM).*
28. Sutandiyo, Weanna;Chhetri, Mohan Baruwal;Loke, Seng Wai;Krishnaswamy,Shonali , Mgaia: Extending the gaia methodology to model mobile agent systems, Proceedings of the 6th International Conference on Enterprise Information Systems (ICEIS), Porto, Portugal, 14-17 April 2004 / I. Seruca, J. Filipe, S. Hammoudi and J. Cordeiro (eds.), pp. 515-518
29. Mohan B. Chhetri, Rosanne Price, Shonali Krishnaswamy and Seng W. Loke, Ontology-based Agent Mobility Modelling, Proceedings of the 39th IEEE Hawaii International Conference on System Sciences – 2006,.
30. PAOLO BRESCIANI AND ANNA PERINI , PAOLO GIORGINI AND FAUSTO GIUNCHIGLIA , JOHN MYLOPOULOS , Tropos: An Agent-Oriented Software Development Methodology , International conference on Autonomous Agents and Multi-Agent Sytems, 8, 203–236, 2004 ,pp-203-236
31. de Hoog, R., Martil, R., Wielinga, B., Taylor, R., Bright, C., & van de Velde, W. (1993) The CommonKADS model set,. ESPRIT Project P5248 KADS-II/M1/DM..1b/UvA/018/5.0, University of Amsterdam, Lloyd's Register, Touche Ross Management Consultants & Free University of Brussels
32. Mikhail Perepletchikov and Lin Padgham , Systematic Incremental Development of Agent Systems, using Prometheus, Proceeding of 5th International Conference on Quality Software (QSIC 2012) , Melbourne, Australia, 19-20 Sept 2005
33. Khanh Hoa Dam , Michael Winikoff, Comparing AgentOriented Methodologies , ACM 089791886/ 97/05.