Available online www.jsaer.com

Journal of Scientific and Engineering Research, 2020, 7(1):1-10

Review Article

ISSN: 2394-2630
CODEN(USA): JSERBR

Prospects, Issues and Challenges of Software Agents in Twenty First Century

Nnebe, S.E.[1], Odighi, M.O.[2], John Otumu, A.M.[1]

[1]Department of Computer Science, Ambrose Alli University, Ekpoma, Nigeria
[2]Department of Computer Science, Federal Polytechnic, Auchi, Edo State, Nigeria

Abstract Software agents are an emerging technology that is making computer systems easier to use by allowing people to delegate work back to the computer. They help do things like find and filter information, customize views of information, and automate work. This paper examines some real agent-enhanced applications to explore the value of agents, summarizes the characteristics that differentiate agents from other software, Mode of operation of software agent, Drawing trends and developments in Information Technology (IT) and Artificial Intelligence (AI) fields, we also discuss on prospects for software agents. Finally, some of the challenges associated with various agents and overview other general issues which pertain to all the types of agents in the typology.

Keywords Agent, Artificial intelligence, Mobile agent, Typology, Mobile agents, Reactive agents

1. What are Software Agents?

A software agent is piece of software that functions as an agent for a user or another program, working autonomously and continuously in a particular environment. It is inhibited by other processes and agents, but is also able to learn from its experience in functioning in an environment over a long period of time. Software agents are computational programs that attempt to model and simulate human mind and human behavior, to become substitutes for human agents. Agents act to accomplish delegated, specialized tasks on behalf of users [1]. They participate actively in accomplishing specific missions by taking initiative autonomously and acting towards reaching certain user-specified or automatically generated goals. Computing programs traditionally depend on users to use them. Software agents make it possible for the programs to work independent of users' presence and instructions, and to deliver only customized, user-wanted information and service. Conventional programs usually remain dormant until specifically called by user instructions, whereas agents are always "alive" and ready for action and do not rely on users' explicit action to be activated.

2. Characteristics of Software agents

- Software agents can do their task without any source intervention
- Social interaction with other software agents and human.
- Software agents are specific in their goals.
- Good software agent is the one which has the attitude to receive and adopt changes (Olofsson,1998).
- The agent must be programmed in a powerful language so as to express the rules. Safety of the information must be promised by the agent.
- Effective usage of the existing resources.
- Agent must be a good sailor.
- Agents must be very careful in handling unauthorized users.

- The same information must be accessed by the user to which they have right.

3. Type of Software agents

Agents are classified into different types based on the characteristics they posses. In order to possess the above properties agents must have distinct features such as locomotion, integration, co-operation, information, stimulation, etc.

Collaborative agents: A collaborative agent is a software program that helps users solve problem, especially in complex or unfamiliar domains by correcting errors, suggesting what to do next, and taking care of low level details.

Interface agents: Interface agents are computer programs that employ machine learning techniques in order to provide assistance to a user dealing with a particular application [2]. These agents take sufficient amount of time to understand and learn human behavior before they are do work. In spite of their artificial learning thoughts they are limited co-operative with other agents [3].

Mobile agents: A mobile agent is an executing program that can migrate during execution from one machine to another in a heterogeneous network [3]. Mobile agents are used to solve many problem of network computing with minimum bandwidth and connectivity. The theme behind these agents is, „give program the ability to move‟. The main advantages of mobile agent over stationery agent are: (a) This is not bound to the system where it begins execution. (b) Can move from one system to another within the network. (c) Both the state and code is transported.

Information/Internet agents: The intelligent part of software which can automatically search for information on the website is termed as information agents. Information system can be considered as knowledge base system. These agents are defined by what they do unlike collaborative agents or interface agents which are defined by what they are [4].

Reactive agents: These agents are responsible for stimulating the response to the present state of the environment in which they are embedded. These agents interact with other agents in a very simple and basic way. The important things which support reactive agents. (a) There is no prior specification of the behavior of these agents set since the dynamic interaction leads to the emergent complexity. (b) Reactive agents are responsible for collection of modules which operate autonomously. (c) Reactive agents tend to operate on representations which are close to raw sensor data [4]. (d) Intelligent behavior is the interaction of these agents with their environment.

Hybrid agents: Combining two or more of the previous mentioned agent philosophies will yield a better functioning.

Heterogeneous agents: These agent systems unlike hybrid, refers to an integrated set up least two or more agents which belong to two or more different agent classes. These may also contain two or more hybrid agents [5].

Deliberative behaviors: Deliberative agents possess an internal reasoning model and exhibits planning and negotiation skills when engaged with other agents in other to achieve their goals. In contrast with deliberative agents, reactive agents lacks an internal reasoning model, but rather act upon the environment using a stimulus response types of behaviors [6].

Smarts agents: The smart agents are the new form of software agents that interface with other agents forming an artificial intelligence. SMART stands for System for Managing Agents for Real Time. The key concept lies here is not the entire individual agent need be intelligent. But by working together in a smart way the agents form a type of emergent intelligence that may appears to exhibit intelligence.

4. Mode of Operation of Software Agent

Each agent is unique in his mode of operation and thus in particular in his program-technical characteristics. The mode of operation of software agent can be generally explained by using a model in figure 1.

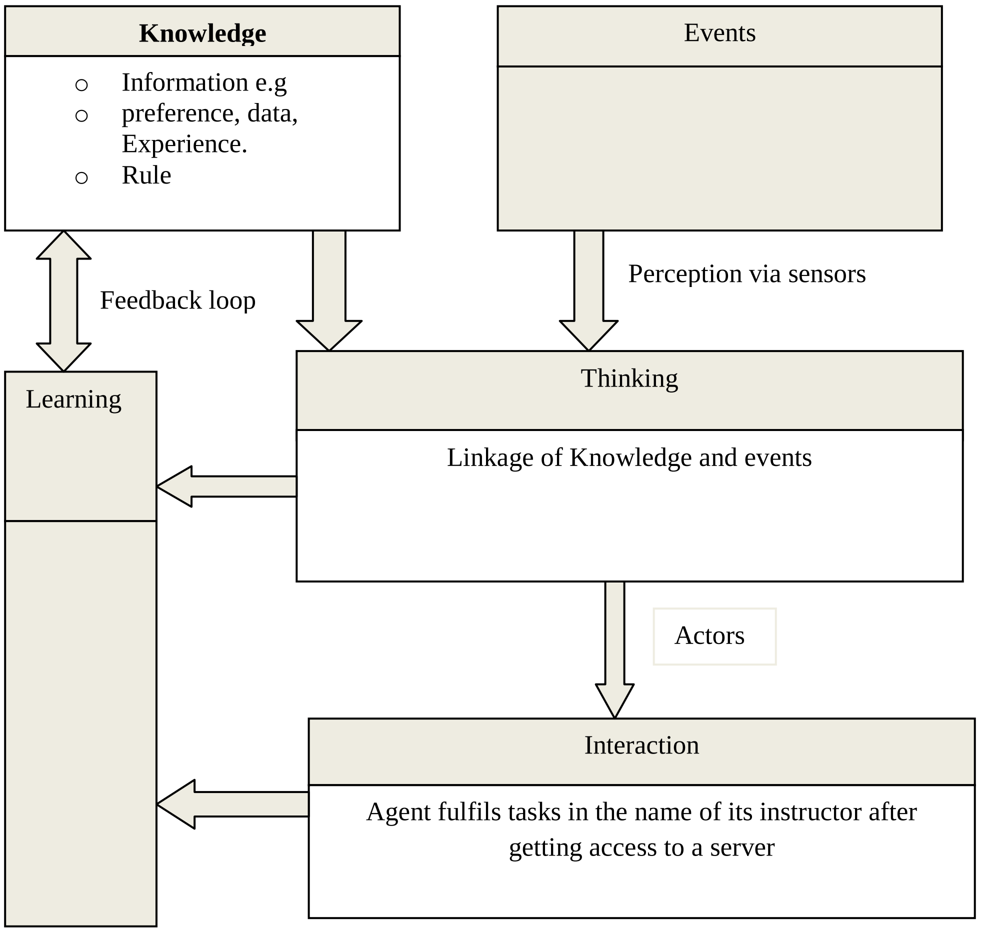

Figure 1: Mode of operation of software agent [5]

The description of the mode of operation of software agents requires first the analysis of the technology, which determines its intelligence. Intelligence can be attributed thereby to three dimensions

- Knowledge
- Thinking
- Learning

The knowledge of an agent consists of information and rules. Rules can consist of simple "if - than" relations or represent also complex neural networks. Thus the agent has an internal knowledge base, which consists of foreknowledge and learned knowledge. By foreknowledge information and rules are to be understood, which are given to the agent in the context of its program structure or are indicated directly by the instructor. Learned knowledge interaction with the environment or conclusions derived through obtained information are to be understood which are based on the foreknowledge [7].

For the use of the knowledge the agent needs the "ability to think", which is called agent machinery. The agent machinery requires two things. On the one hand the agent must assume events in its environment over sensors. On the other hand it must combine the assumed events with its knowledge in a "thought process"[8].

From this linkage the agent can draw conclusions. From the conclusions the agent can initiate autonomously, thus without intervention or statements of the instructor, and act through its actors. For executing an action the agent must, in some cases, have an access authorization. If the access authorization is given, then the agent can execute its task, by interacting with humans, computers or other software agents. The intelligence of the agent increases by the enlargement of its knowledge base. By extending the available information and rules the agent can draw better conclusions from events in its environment. The extension of the knowledge base results from learning processes. Learning is a behavior modification, which is the result of an experience.

A software agent learns using his learning machinery by the following foreknowledge information and rules are to be understood, which are given to the agent in the context of its program structure or are indicated directly by the instructor. By learned knowledge interaction with the environment or conclusions derived through obtained information are to be understood which are based on the foreknowledge.

- From the linkage of knowledge and events in the agent machinery develop conclusions, which enter the knowledge base over the learning machinery.
- The interaction with the environment activates further learning processes. These learning processes direct essentially at adding or changing rules or information.
- Also without interaction or reaction to events the agent can learn.

With respect to the literature further distinctions are made between stationary and mobile. Mobile agents execute their tasks, by copying themselves within a network from computer to computer, in order to execute the specific function in each case "locally". It is necessary to remark that the mobility or non-mobility of an agent is in the reason only two different technical types of the same concept. Therefore, no substantial differences for the instructor of the agent arises, although this is often denied. Besides mobile agents directly raise safety questions, since permitting in principle "stranger" agents on a computer often also opens the doors for hackers. The boundaries between hostile mooded mobile agents and viruses are flowing [9].

5. Prospects for Software Agents

In this Section we speculate on plausible trends for the development of software agents in twenty first century years. We first look at developing trends in the underlying technologies and application fields. We then discuss the likely nature of their impact on agent development.

Trends in Information Technology (IT): Hardware, Software, Digitizing, Coordinating, Informating and Interfacing

In foreseeing the future of information technology, Grover and Segars [10] emphasized four important trends: (i) hardware is evolving in the direction of smaller and more powerful machines; (ii) software is evolving in the direction of user-friendly, modular and flexible platforms; (iii) cost of digitizing is approaching zero; (iv) cost of coordinating, which refers to all costs involved in acquiring and evaluating information, is also approaching zero. Cost of digitizing approaching zero means that every single piece of information interpreted by our brain through our five senses can be represented in digital form at no cost. The cost of coordinating approaching zero means that there will be a lot of opportunity for innovation in companies and for securing better services by customers [10]. These will cause further changes in the way of supplying information and interfacing of software systems. Information supply will be very different from the current "react to needs and provide" cycle. Smaller but powerful processing technologies that allow information consumers to access, manipulate and communicate different forms of information will be needed [10]. At the same time, it is also likely that information itself will contain processing capabilities and become smart data that can recognize complementary form of data which can be used to form larger information constructs useful to information consumer [10].

5.1.1. IT Management - Information Focused Culture

Markus [11] developed insights into the future of IT management through a hypothetical case in which IT recession happened. He speculated that "perhaps the most common new IT management role in the organizations of the future will be IT-use facilitators who work closely with people to help them learn new tools and discover new work methods". Then, when users became familiar with technology, spent less time dealing with technology problems, had reliable system and infrastructure, the focus will move from understanding about technology to the effective use of information. Computers came to be viewed as information appliances. In many ways IT in the early 21st century became like automobiles - simply ways to get things done. Later on in the 21st century (2020) IT will entirely be taken for granted as does electricity today [11]. Organizations will develop information-focused culture. The attention of IT management will be paid to information consumers on how to identify users' information needs, how information could be sought, acquired and used to improve business decisions. The ability to quickly evolve down the data-information-knowledge-wisdom-action cycle will be the most important asset for organizations. On one hand, companies may engage in large scale data warehousing and data mining of warehouses combining internal and external data. On the other hand, many entrepreneurial investor resources are being devoted to creating and consolidating external Databases and content archives. They have strong incentives to try to draw user organizations away from computing and managing their own Databases. Third, advances in communications and processing technologies may render collective storage obsolete. Information may be coded to remain with its source, only to be accessed or aggregated when needed [11]. Interestingly, Davenport [12] also described a future environment for information systems in which information replaces technology as the primary focus of a firm's efforts. Schermer [13] then described the future role of information professionals as a technical knowledge and business information broker who knows where to look for information, how to screen it and interpret it, how to utilize it internally for specific purpose to create knowledge and expertise within their organization.

5.1.2. Personalized Service

The development in information and communication technology also incur changes in the way of service supplying. Schermer [13] believed that the information age will cause a dramatic change in the form, extent and delivery of service. IT will create a society in which services are tailored to the personal tastes of customers. Similar to the agricultural technology revolution which brought plentiful and inexpensive food, and the industrial technology revolution which brought plentiful and inexpensive goods, information technology revolution will bring many highly personalized services. "It will enable the great majority of citizens to receive a level of customized service now enjoyed only by the effluent" [6].

5.2. Trends in Artificial Intelligence Applications:

Artificial Intelligence (AI) is concerned with programs that respond flexibly in situations that were not specifically anticipated by the programmer. To make computers/programs smarter and smarter and come closer and closer to imitating, or ultimately surpassing, complex human thought processes, we need to draw upon advances in artificial intelligence.

Robotics has always been a branch of AI research and application. Robots were originally conceived to do tediously repetitive assembly-line tasks. Over the years, however, it has become apparent that in addition to never getting bored, never getting sick and never going on strike, robots can be designed to work far more rapidly and precisely than human alone. Nevertheless, it has been a challenge to equip them with enough sensory perception, computing power and real-time response capability to adapt to such situation as having a part presented slightly askew on an assembly line. As computer components have gotten exponentially smaller and algorithms have gotten exponentially more clever, robots have become more and more capable and flexible in their behavior [7].

In the centuries to come, robots will play crucial roles in exploring the space. Software agents are soft robots, or softbots. They represent one integral part of AI too. Their development naturally takes great advantage of advances in other AI branches, while they at the same time motivate and justify that research. As Esther Dyson predicted long time ago that AI would not become truly important commercially until AI becomes embedded in main-stream, strategically important systems like raisins in a loaf of raisin bread, time has proven Dyson's prediction correct. AI as a field is becoming more important as emphasis shifts away from replacing expensive human experts with stand-alone expert systems toward enabling or enhancing main-stream computing systems that create strategic advantage. Accordingly, many of today's AI systems are connected to large data bases, deal with legacy data, talk to networks, handle noise and data corruption with style and grace, implemented in popular languages, and run on standard operating systems [14].

AI will certainly be an essential part, although often a small part, of the total solution to customers' problems. AI technology will go beyond expert systems, where computers could do what human experts could do, only less expensively. The emphasis will move from doing what people do to exploiting opportunities to do tasks that people cannot do alone. The commercial value of AI will lie in the direction of new revenue building rather than replacing people. AI will play significant roles in offering new capabilities and creating new revenue streams in areas such as operating systems, interfaces and information access, indicated by AI enabled information access, AI enabled human-computer interaction, and AI driven advances in computing infrastructure. Through truly gigantic high-resolution displays that are coming, more natural and familiar modes of communication that may soon make the keyboard obsolete, all rely on AI technology. To provide help to WWW-users in information access, the creators of aggregation tools must take advantage of AI work [7].

AI research has developed new ways of analyzing heaps of data that complement traditional statistical methods. AI technology can take us a great leap forward from systems limited to key word analysis. Software agents, like Bayes nets, neural nets and genetic algorithms, is regarded as newer technical ideas in the field of AI. All of them are important; but none, by itself, is the answer. When combined with older ideas such as rule chaining, they will form a powerful unit. In fact, the power of software agents lie exactly in their synthesized use of multiple AI techniques and integration with mainstream business applications [9].

Kendall [15] described the development of AI systems as will follow four paradigms: (i) cognitivist (based on symbolic techniques, with applications such as expert systems, knowledge based systems). (ii) connectionist (neural nets). (iii) action-selection paradigm (represented by autonomous agents selecting actions that emerge from the interaction of multiple, diverse, relatively independent modules), and (iv) evolutionary paradigm (genetic algorithm, evolving systems that can access the merits of inconsistent motives, can store and compare motives, change upon self-evaluation, and adapt to approval or disapproval of external agents).

According to Kendall [15], the evolutionary paradigm will be in store for AI for a future of perhaps 50 years. In ten years, however, our focus will be on autonomous agents of the action-selection paradigm. Autonomous agents will evolve from being just capable of autonomous, purposeful actions that responds to external stimuli, to intelligent, adaptive agents that are capable of performing actions they were not capable of previously by acquiring knowledge and building on earlier actions [15].

5.3. Software Agents in twenty first century

The benefits of using software agents include speed, productivity, customized service, new way of information and support supplying, among others. In this section we relate the future of software agents to the above-mentioned developing trends in IT and AI.

The Needs

Changes in the way of information supplying call for software agent applications. Software agents do not follow only the "react to needs and provide" cycle. They can actively push the required information to users. They can also bring a more rich and novel information environment to users. The needs for smaller but powerful processing technologies to allow consumers of information to access, manipulate and communicate information, actually proposes a heavy processing load for computer programs. They must have agent-units embedded in or have multiple agents to collaborate on. Finally, data itself becoming smart also relies on the embedding of agent-like processing constructs. Information focused culture will lay great emphasis on organizations' competence in quickly discovering knowledge from large volumes of data sets. Knowledge discovery from electronic data in multiple formats is already, and will still be an important area of intelligent agent applications. In cases where advances in communications and processing technologies may result in collective storage becoming obsolete and data will remain with their sources, agents will be needed to visit dispersed sources, to extract needed information, to synthesize and analyze the data and to add value. A society offering highly customized services would create a huge demand for service staff. Service providers or companies would try to employ technology and systems instead of just throwing more service at a problem, This, however, will lay the burden on customers. Software agents can be generated to serve the customers. Agents as service providers take the human element out of service thereby eliminating variability. They enable customization by allowing IT to create and deliver the precise service that the customer desires. Software agents actually translate a service delivery strategy into an effective IT infrastructure [7]. To summarize, agent technology can make computer applications more efficient and effective. They bring competitive benefits to software systems [16] and further, in business operations. The software agent market place is far from mature, and it is set for rapid growth and owns a huge potential.

6. Applications and Benefits of Software Agents

a. Agents make less work for the end user and application developer.
b. The agent can adapt to its user preferences and habit over a course of time.
c. It will intelligently get shared among the community.
d. Mobile agents manage the user‟s E-mail, fax, phone and pager as well as linking the user to Telescript enabled messaging and communication services such as America Online and AT&T Persona Link Services [16].
e. The most favorite area with respect to reactive agents is games and entertainment industry.
f. Shopping agents are ideal applications of Agent Builder agents. These agents can be used to locate [17] merchandise, compare prices, place orders, etc.

7. General Issues of Software Agents

We have outlined their various promises as well as their challenges. However, there are issues which society would have to grapple with through various legislations and they would be very thorny. They include the following:

• **Privacy:** how do you ensure your agents maintain your much needed privacy when acting on your behalf?

• **Responsibility which goes with relinquished authority**: when you relinquish some of your responsibility to software agent(s), be aware of the authority that is being transferred to it/them. How would you like to come back home after a long hard day at work being the proud owner of a used car negotiated and bought for, courtesy of one of your software agents? How do you ensure the agent does not run up a huge credit card bill on your behalf?

• **Legal issues**: following on from the latter, imagine your agent offers some bad advice to other peer agents resulting in liabilities to other people, who is responsible? The company who wrote the agent? You who customized it? or Both? We envisage a new raft of legislation would need to be developed in the future to cover software agents.

• **Ethical issues:** these would also need to be considered. Already, Norman [18] is already concerned enough about the ethics of software agents that he has proposed an agent etiquette for information service and user agents as they gather information on the WWW.

8. Challenges of Software Agents

The problem arises where the two parties have different interests and asymmetric information (the agent having more information), such that the principal cannot directly ensure that the agent is always acting in their best interest, particularly when activities that are useful to the principal are costly to the agent, and where elements of what the agent does are costly for the principal to observe. Often, the principal may be sufficiently concerned at the possibility of being exploited by the agent that they choose not to enter into the transaction at all, when it would have been mutually beneficial: a suboptimal outcome that can lower welfare overall [19].

The agency problem can be intensified when an agent acts on behalf of multiple principals. When one agent acts on behalf of multiple principals, the multiple principals have to agree on the agent's objectives, but face a collective action problem in governance, as individual principals may lobby the agent or otherwise act in their individual interests rather than in the collective interest of all principals. As a result, there may be free-riding in steering and monitoring, duplicate steering and monitoring, or conflict between principals, all leading to high autonomy for the agent. This has been coined the multiple principal problem and is a serious problem in particularly the public sector, where multiple principals are common and both efficiency and democratic accountability are undermined in the absence of salient governance.

Challenges of mobile Agents include the following:

- Transportation: how does an agent move from place to place? How does it pack up and move?
- Authentication: how do you ensure the agent is who it says it is, and that it is representing who it claims to be representing? How do you know it has navigated various networks without being infected by a virus?

- Secrecy: how do you ensure that your agents maintain your privacy? How do you ensure someone else does not read your personal agent and execute it for his own gains? How do you ensure your agent is not killed and its contents 'core-dumped'?
- Security: how do you protect against viruses? How do you prevent an incoming agent from entering an endless loop and consuming all the CPU cycles?
- Cash: how will the agent pay for services? How do you ensure that it does not run amok and run up an outrageous bill on your behalf? In addition to these are the following:
- Performance issues: what would be the effect of having hundreds, thousands or millions of such agents on a WAN?
- Interoperability/communication/brokering services: how do you provide brokering/directory type services for locating engines and/or specific services? How do you execute an agent written in one agent language on an agent engine written in another language? How do you publish or subscribe to services, or support broadcasting necessary for some other coordination approaches?

Challenges of Reactive Agents include the following:

- Expanding the range and number of applications based on reactive agents;
- Methodology: there is a yearning need for a clearer methodology to facilitate the development of reactive software agent applications. This may or may not require the development of more associated theories, architectures and languages. Much of the current approaches, sadly, smacks of 'trial and error'
- Non-functional issues: issues such as scalability and performance would need to be addressed, though these are unlikely to be important until clearer methodologies have been developed and evaluated.

Challenges of Interface Agents include the following:

- Demonstrating that the knowledge learned with interface agents can truly be used to reduce users' workload.
- Carrying out hundreds of experiments using various machine learning techniques over several domains to determine which learning techniques are preferable for what domains and why;
- Analyzing the effect of the various learning mechanisms on the responsiveness of agents;
- Extending interface agents to be able to negotiate with other peer agents;
- Enhance continually the competence of interface agents so that their users' trust in them build up over time [20].

Challenges of coordination agents includes the following:

- Inter-agent coordination, this is a major issue in the design of these systems. Coordination is essential to enabling groups of agents to solve problems effectively. Without a clear theory of coordination, anarchy or deadlock can set in easily in collaborative agent systems? Furthermore, should agents be totally truthful when negotiating with others or should they be allowed to 'lie' when it suits them? Coordination is also required due to the constraints of resource and time. Much experimental and/or formal work is still required to address these issues of coordination and negotiation [19].
- Stability, Scalability and Performance Issues: these issues have yet to be acknowledged, yet alone tackled in collaborative agent systems research. Empirical investigations need to be carried out to establish suitable minimum levels of performance and, clearly, these systems have to be stable. Alternatively, their stabilities would need to be proven formally. Though, these issues are non-functional, they are crucial nonetheless;
- Legacy systems: the thorny issue of what to do with legacy systems is still with us and will always be a problem. Established techniques and methodologies for integrating agents and legacy systems are still required;

- How do these systems learn? Would learning not lead to instability? What are the appropriate architectures for different types of problems? How do you ensure an agent does not spend much of its time learning, instead of participating in its set-up?
- Evaluation of collaborative agent systems: this problem is still outstanding. How are they verified and validated to ensure they meet their functional specifications? Are unanticipated events handled properly? How else would you trust such systems to run power stations, nuclear installations and chemical plants.

9. Conclusion

In this paper, we have tried to convey some of the key concepts, characteristics, applications and mode of operation of the vibrant field of software agents different conclusions were drawn that software agents are flexible enough to adopt and adjust themselves into changes. Software agents provide security to the information. Software agents are currently being used in hundreds of applications, both to solve new types of problems (such as personal information management and electronic commerce) and more traditional problems (such as business process management and network management). However, Software agents will not be all pervasive. They will not magically solve all the difficult problems which exist in the current generation of advanced information processing systems e.g planning in uncertain environments, perceiving and acting in a timely fashion in response to environmental changes, and inferring a user's preferences based on their behaviour. Moreover, by their very nature software agents create a new set of problems which must be tackled. Because they are autonomous, users may be wary in trusting them to act on their behalf

References

[1]. Wooldridge, M.; Jennings, N. R. (1996). "Intelligent agents: theory and practice". 10 (2). Knowledge Engineering Review: 115–152

[2]. Cabri, G., Leonardi, L., and Zambonelli, F. (1998), "Mobile Agent Technology: Current Trends and Perspectives", Zambonelli, Congresso AICA, Napoli, 12 pp.

[3]. Olofsson S. (1998), "Scan Agent: Ett Intelligent Datainsamlingsverktyg", Master thesis, IAMSR Ser. A: 484, Åbo 1998.

[4]. Wooldridge, M. & Jennings, N. (1995), Intelligent Agents, Lecture Notes in Artificial Intelligence 890, Heidelberg: Springer Verlag.

[5]. Nwana, H. S. and Wooldridge, M. (1996), "Software Agent Technologies", British Telecommunications Technology Journal 14 (4), October.

[6]. Burkle, A., Essendorfer, B., Hertel, A., Muller, W., and Wieser, M. (2006), "A Test Suite for the Evaluation of Mobile Agent Platform Security*", Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology, 5pp.*

[7]. Charles H. (2016). Comprehensive Introduction to Intelligent Software Agents for Professional Accountants, CPA (Charles.Hoffman@me.com) August 16, 2016 (DRAFT).

[8]. Nwana, H. S. (1996). "Software Agents: An Overview". Knowledge Engineering Review. 21 (3): 205–244.
CiteSeerX 10.1.1.50.660. doi:10.1017/s026988890000789x.

[9]. Shuhua L. (1999). Perspectives on and Prospects for Software Agent Systems, Åbo Akademi University Institute of Advanced Management Systems Research Lemminkäisenkatu 14, FIN-20520 Turku, Finland. Turku Centre for Computer Science TUCS Technical Report No 269.

[10]. Grover V. and A. H. Segars A. (1998). "IT: the Next 110010 Years", The DATABASE for Advances in Information Systems, Fall, Vol. 27, No. 4, pp. 45-57.

[11]. Markus M. L., (1996) "The Future of IT Management", The DATABASE for Advances in Information Systems, fall, Vol. 27, No. 4, pp. 68-84.

[12]. Davenport T. H. (1996). "The Second Information Revolution", The DATABASE for Advances in Information Systems, fall, Vol. 27, No. 4, pp. 85-91.

[13]. Schermer, B. W. (2007). Software agents, surveillance, and the right to privacy: A legislative framework for agent-enabled surveillance (paperback). Leiden University Press. pp. 140, 205–244. hdl:1887/11951. ISBN 978-0-596-00712-6. Retrieved 2012-10-30.

[14]. Mueller, J. P. and Tambe, M. (2010), Intelligent Agents II, Lecture Notes in Artificial Intelligence 1037, Heidelberg: Springer Verlag.

[15]. Kendall K. E., (1996)"Artificial Intelligence and Götterdämerung: The Evolutional Paradigm of the Future", The DATABASE for Advances in Information Systems, fall, Vol. 27, No. 4, pp. 99- 115.

[16]. King D. and K. Jones, (1995). "Cpmpetive Intelligence, Software Robots and the Internet: The News Alert Prototype", Proceedings of the HICSS'28 Conference, IEEE Computer Society Press, Los Alamitos, CA, 1995, pp. 624-631.

[17]. Singh S. K., and Watson R., (2001). "Technology, Task, People and Structure: An Examination of their interdependencies in the context of an EIS", Journal of Decision Systems. Vol. 3. No. 4, 1994, pp. 359-383.

[18]. Norman D. A., (2000) "How Might People Interact with Agents", Communications of the ACM, Vol. 37, No. 7, pp. 68-71.

[19]. Agre, P. E. (1988), The Dynamic Structure of Everyday Life, Ph.D Thesis, Department of Electrical Engineering and Computer Science, MIT.

[20]. Chang, D. and Lange, D. (1996). Mobile Agents: A New Paradigm for Distributed Object Computing on the WWW, OOPSLA'96 Workshop, 8pp.