International Journal of Information Technology and Knowledge Management
July-December 2010, Volume 2, No. 2, pp. 251-254

A FRAMEWORK ON TYPOLOGY OF SOFTWARE AGENTS

Meenakshi[1], Mona Sehgal[2] & Payal Anand[3]

Software agent is a extremely developing area of research. Within framework of software agents extensive research work has been carried out. The main goal of this paper is to reanalyze the rapidly evolving area of software agents. Firstly, this paper presents a definition, attributes and scope of software agents and then presents a typology of agents. Next, it presents critique view of software agents. It also spells out some other general issues and future of the agents. Finally, we conclude with brief description of MAS, a real time software agent system.

1. Introduction

An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators.

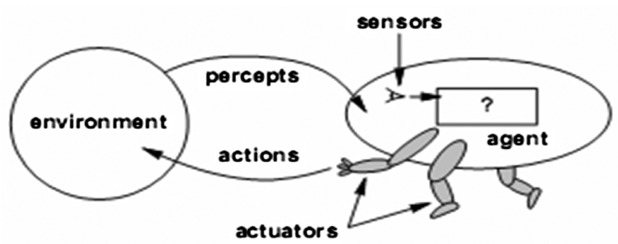

Fig. 1: Software Agent and Environment

A percept is the agent's perceptual inputs at any given instant. A percept sequence is the complete history of everything the agent has ever perceived. In general, an agent's choice of action at any given instant can depend on the entire percept sequence observed to date.

An agent's behavior is described by the agent function that maps any given percept sequence to an action.

"Agent is that agent does"[1] is a slogan that captures, albeit simplistically, the essence of the insight that agency cannot ultimately be characterized by listing a collection of attributes but rather consists fundamentally as an attribution on the part of some person.[2]

Consistent with the requirements of a particular problem, each agent might possess to a greater or lesser degree attributes like:

- Reactivity: The ability to selectively sense and act.
- Autonomy: Goal-directedness, proactive and self-starting behavior
- Collaborative Behavior: Can work in concert with other agents to achieve a common goal.
- "Knowledge-level" Communication Ability: The ability to communicate with persons and other agents with language more resembling human like "speech acts" than typical symbol-level program-to-program protocols.
- Inferential Capability: Can act on abstract task specification using prior knowledge of general goals and preferred methods to achieve flexibility; goes beyond the information given, and may have explicit models of self, user, situation, and/or other agents.
- Temporal Continuity: Persistence of identity and state over long periods of time.[3]
- Personality: The capability of manifesting the attributes of a "believable" character such as emotion.
- Adaptivity: Being able to learn and improve with experience.
- Mobility: Being able to migrate in a self-directed way from one host platform to another.

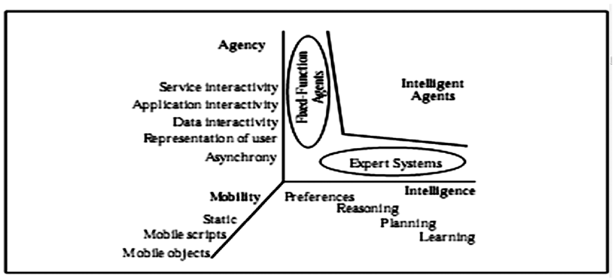

Fig. 2: Scope of Intelligent Agents [4]

[1]Lecturer, Deptt. of MCA, GNKITMS, Yamuna Nagar, Haryana, India.

[2]Lecturer, Deptt. of MCA, GNKITMS, Yamuna Nagar, Haryana, India.

[3]Lecturer, Deptt. of MCA, GNKITMS, Yamuna Nagar, Haryana, India

E-mail: [1]minisingh2@yahoo.in, [2]mona_sehgal@rediffmail.com, [3]payalanand79@gmail.com

An influential white paper from IBM (Gilbert et al. 1995) described intelligent agents in terms of a space deûned by the three dimensions of agency, intelligence, and mobility (Fig.2)[4]:

Agency is measured qualitatively by the nature of interaction between agent and other entities like data, applications, services etc. in the system. Agency is the degree of autonomy and authority vested in the agent. Intelligence is measured both in terms of the user's objectives, and in terms of the resources available to the agent. Intelligence is the degree of reasoning and learned behavior i.e. the agent's ability to accept the user's statement of goals and carry out the task delegated to it. Mobility is the degree to which agents themselves travel through the network.

2. Typology of Agents

A typology can be defined as the study of types of various entities. In this section, we investigate a typology of agents and categorize software agents in different classes. There are various aspects to classify prevailing software agents.

Firstly, agents may be classified by their mobility, i.e. by their ability to move around some network. This yields the classes of static or mobile agents. Secondly, they may be classed as either deliberative or reactive. Deliberative agents possess an internal symbolic, reasoning model. They engage in planning and negotiation in order to achieve coordination with other agents. On the other hand, Reactive agents don't possess any internal, symbolic models. They act using a stimulus/ response type of behaviour by responding to the present state of the environment in which they are embedded (Ferber, 1994)[5]. Work on reactive agents originate from research carried out by Brooks (1986) [6] and Agre & Chapman (1987) [8]. Brooks has argued that intelligent behaviour can be realised without the sort of explicit, symbolic representations of traditional AI (Brooks, 1991b) [7].

Thirdly, agents may be classified along several ideal and primary attributes which agents should exhibit i.e. autonomy, learning and cooperation. Autonomy refers to the principle that agents can operate on their own without the need for human guidance. (Wooldridge & Jennings, 1995a) [9]. Cooperation with other agents is paramount. In order to cooperate, agents need to possess a social ability. (Wooldridge & Jennings, 1995a) [9].

We use these three minimal characteristics in Fig.1 to derive four types of agents to include in our typology:

Truly smart agents do not yet exist: indeed, as Maes (1995a) [11] notes "current commercially available agents barely justify the name", yet alone the adjective 'intelligent'.

In principle, by combining the two constructs so far (i.e. static/mobile and reactive/deliberative) in conjunction with the agent types identified (i.e. collaborative agents, interface agents, etc.), we could have static deliberative collaborative agents, mobile reactive collaborative agents, static deliberative interface agents, mobile reactive interface agents, etc. But these categories, though quite a mouthful, may also be necessary to further classify existing agents. For example, Lashkari et al. (1994) [12] presented a paper at AAAI on 'Collaborative interface agents' which, in our classification, translates to static collaborative interface agents.

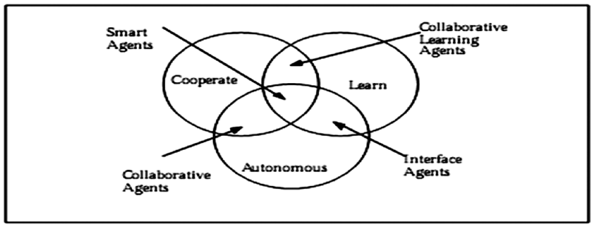

Fig. 3: Typology of Agents [10] Primary Attribute Dimension

Fourthly, agents may sometimes be classified by their roles (preferably, if the roles are major ones), e.g. World Wide Web (WWW) information agents. Fifthly, we have also included the category of hybrid agents which combine of two or more agent philosophies in a single agent. Lastly, for agent systems to be truly 'smart', they would have to learn as they react and/or interact with their external environment.

Hence, we identify seven types of agents:

- Collaborative agents;
- Interface agents;
- Mobile agents;
- Information/Internet agents;
- Reactive agents;
- Hybrid agents;
- Smart Agents.

3. A Critique of Typology

There are always two faces of the same coin. The above classification of agent typology is somewhat subjective. On one hand collaborative agents and interface agents are defined by what they are but on other side information agents are defined by what they do. So they can't be grouped together.

Interface agents are collaborative agents implemented using reactive technology. Hence, we do not agree fully with the assertion that mobile agents, reactive agents and hybrid agents are all underlying technologies for implementing the former classes. Reactive agents for example, have a distinct philosophy, hypothesis, etc. which make it stand out from the rest. Mobility is not a necessary condition for agenthood, we agree with the statement. As agents exist in a multidimensional space, and for clarity, we have merged this multi-dimensional space into a single list.

We described a 'flat' structure for agent classification because of simplicity, clarity and generality as hierarchical structure seems to be less clear. Last but not least, our typology highlights the key contexts in which the word 'agent' is used in the software literature.

4. SOME GENERAL ISSUES AND THE FUTURE OF AGENTS

Agents are posing a no. of social, legal and ethical challenges. The main issue for agent technology is that how society will react and adapt agent technology. Society would have to adapt this technology through various legislations and they would be very irony.

Social issues: Society generally includes the following:

- Privacy: Acting on the behalf of the user how agents would ensure the required privacy of the user.
- Responsibility: How agents would use the delegated authority to accomplish the given task to escape the unintended side-effects.

Legal Issues: Legislations would need to be developed in the future to cover software agent's liabilities.

Ethical Issues: What would be the agent ethics needed to be considered?

These issues include identity, information search, authorization, accurate and updated services, limitations etc. with regard to agent's perspective.

Etzioni & Weld (1994) [13] have proposed some ethical issues:

- Safety: The agent should not destructively alter the world;
- Tidiness: The agent should leave the world as it found it;
- Thrift: The agent should limit its consumption of scarce resources;
- Vigilance: The agent should not allow client actions with unanticipated results.

Agent technology is not a passing trend. In this technological era agent technology is broadly used. Researchers are further exploring the use of agent technology. Agents would leave a considerable impact in day-to-day life. Probably Agents would initially leverage simpler technologies available in most applications (e.g. word processors, spreadsheets or knowledge-based systems). Then agents would gradually be evolved into more complicated applications. Software agents are just about to 'crawl' out of research laboratories and there is still a very long way to go.

5. REAL TIME SOFTWARE AGENT BASED SYSTEM

For working in real time environment more sophisticated agent system is needed to be used. There are various methodologies available worldwide according to application requirement. e.g. MAS. A distributed Multi-Agent System (MAS) is designed to establish a fault-tolerant system behavior. MSS is a monitoring concept based on software agents to provide comprehensive data for an accurate assessment of the lifespan of the system. A novel design methodology is used for creating agent-based monitoring systems. The MAS is linked via a network connection. As a result, exchangeability of each subsystem is achieved. By introducing independent subsystems, the overall system architecture is in harmony with current monitoring standards which recommend a client/server-like pattern as global system architecture (IEC 61400-25, [14-16]).

6. CONCLUSION

This paper concludes the broad literature of software agents and then classifies the agents in different categories according to specific criteria. Various pros and cons are discussed of given typology and other general issues of software agents. We have reanalyzed the software agent's framework and we can explore much more in this area by using recent technologies.

REFERENCES

[1] Ball, G. 1996. Lifelike Computer Characters (LCC-96) Schedule and Workshop Information. http://www.research.microsoft.com/lcc.htm.

[2] Ball, G.; Ling, D.; Kurlander, D.; Miller, J.; Pugh, D.; Skelly, T.; Stankosky, A.; Thie,D.; Dantzich, M. V; and Wax, T. 1996. "Lifelike Computer Characters: The Persona Project at Microsoft Research". In Software Agents, ed J. M. Bradshaw. Menlo Park, CalifAAAI Press.

[3] Boy, G. 1992. "Computer-Integrated Documentation". In Sociomedia: Multimedia, Hyper-media, and the Social Construction of Knowledge, ed. E. Barrett, 507–531. Cambridge, Mass.: MIT Press.

[4] Gilbert, D.; Aparicio, M.; Atkinson, B.; Brady, S.; Ciccarino, J.; Grosof, B.; O'Connor, P.; Osisek, D.; Pritko, S.; Spagna, R.; and Wilson, L. 1995. IBM Intelligent Agent Strategy, IBM Corporation.

[5] Ferber, J. (1994), "Simulating with Reactive Agents", In Hillebrand, E. & Stender, J. (Eds.), Many Agent Simulation and Artificial Life, Amsterdam: IOS Press, 8-28.

[6] Brooks, R. A. (1986), "A Robust Layered Control System for a Mobile Robot", IEEE Journal of Robotics and Automation, 2 (1), 14-23.

[7] Brooks, R. A. (1991b), "Intelligence without Representation", Artificial Intelligence, 47, 139-159.

[8] Agre, P. E. & Chapman, D. (1987), "Pengi: An Implementation of a Theory of Activity", Proceedings of the 6th National Conference on Artificial Intelligence, San Mateo, CA: Morgan Kaufmann, 268-272.

[9] Wooldridge, M. & Jennings, N. (1995a), "Intelligent Agents: Theory and Practice", The Knowledge Engineering Review 10 (2), 115-152.

[10] Nwana, H. S. 1996. Software Agents: An Overview. Knowledge Engineering Review,11(3): 205-244.

[11] Maes, P. (1995a), "Intelligent Software", Scientific American 273 (3), September.

[12] Lashkari, Y., Metral, M. & Maes, P. (1994), "Collaborative Interface Agents", In Proceedings of the 12th National Conference on Artificial Intelligence, 1, Seattle, WA, AAAI Press, 444-449.

[13] Etzioni, O. & Weld, D. (1994), A Softbot-Based Interface to the Internet", Communications of the ACM 37 (7), 72-76.

[14] International Standard IEC 61400-25 (All Parts). 2006. Communications for Monitoring and Control of Wind Power Converters, TC 88. 2006-12. International Electrotechnical Commission, Geneva, Switzerland.

[15] Giebel, G., Gehrke, O., McGugan, M., and K. K. Borum. 2006. "Common Access to Wind Turbine Data for Condition Monitoring. The IEC 61400-25 Family of Standards," in Proceedings of the 27th Risø International Symposium on Materials Science, Risø National Laboratory, Denmark.

[16] Giebhardt, J., et al. 2004. "Predictive Condition Monitoring for Offshore Wind Energy Converters with Respect to the IEC 61400-25 Standard," in Proceedings of the International Technical Wind Energy Conference (DEWEK), Wilhelmshaven, Germany.