

Agent-based Web Engineering

J. M. Corchado¹, R. Laza², L. Borrajo², J. C. Yañez², A. de Luis¹ and M. Glez-Bedia¹

¹Departamento de Informática y Automática, University of Salamanca, Salamanca, Spain
corchado@usal.es

²Departamento de Informática, University of Vigo, 32004, Ourense, Spain

Abstract. Technological evolution of the Internet world is fast and constant. Successful systems should have the capacity to adapt to it and should be provided with mechanisms that allow them to decide what to do according to such changes. This paper shows how an autonomous intelligent agent can be used to develop web-based systems with the requirements of today users. Internet applications should be reactive, proactive, and autonomous and have to be capable of adapting to changes in its environment and in the user behavior. The technological proposal presented in the paper also facilitates the interoperability and scalability of distributed systems.

1 Introduction

Internet is growing and evolving continuously. Web pages and applications need to be updated and most of the web sites are dynamic. Such systems need to be redesigned frequently and adapted to the technological changes and requirements. Successful web-based systems should have the capacity to adapt to changes automatically and should be provided with mechanisms that allow them to decide what to do according to environmental and user changes. Such mechanisms are known as autonomous or intelligent agents [21]. This paper presents an autonomous agent architecture that may be used to construct dynamic, scalable, intelligent, autonomous, reactive, and proactive Web-based systems.

Agents and multiagent systems have evolved substantially in the last decade in parallel to the Internet explosion and expansion. In most computing systems, all the executed actions are previously planned and encoded by a programmer [2]. However, in our present-day world, where technological evolution is fast and constant, it is necessary to build up systems with the capacity for adaptation and provided with mechanisms that allow them to decide what to do according to their objectives. Such systems are known as agents [26]. This paper shows how to build automatically autonomous agents for the Internet, using a case-based reasoning (CBR) system. The proposed method facilitates the automation of their construction and provides them with the capacity of learning and therefore of autonomy. These agents will facilitate the construction of dynamic and intelligent application for the Internet.

Autonomous agents and multiagent systems should be able to reply to events, which take place in their environment, to take the initiative according to their goals, to

interact with other agents (even human), and to use past experiences to achieve present goals. These properties may be very beneficial to Internet applications. There are different types of agents and they can be classified in different ways [27]. One way is the so-called deliberative agents with a BDI architecture, which have mental attitudes of Beliefs, Desires and Intentions. In addition, they have the capacity to decide what to do and how to get it according to their attitudes [15], [20], [26]. The paper describes how intelligent Internet systems may be constructed with autonomous agents that reason with the help of a CBR system [8], [14].

This paper reviews first the concept of autonomous deliberative agents. Then the proposed model is outlined, and it is shown how a CBR system is used to operate the mental attitudes of a deliberative agent. Several Internet and Wap applications have been constructed with this technology and this paper presents a case study: a sales support e-business system. Finally the proposal is evaluated and some conclusions are exposed.

2 Autonomous agents for the Internet

As mentioned above, autonomous deliberative agents, with a BDI architecture, are composed of beliefs, desires and intentions. The beliefs represent their information state, what the agents know about themselves and their environment. The desires are their motivation state, what the agents are trying to achieve, and the intentions represent the agents' deliberative states. Intentions are sequences of actions; they can be identified as plans. These mental attitudes determine the agent's behaviour and are critical to attain proper performance when the information about the problem is scarce [2], [17]. A BDI architecture has the advantage that it is intuitive, and it is relatively simple to identify the process of decision-making and how to perform it. Furthermore, the notions of belief, desires and intentions are easy to understand. On the other hand, its main drawback lies in finding a mechanism that permits its efficient implementation. The formalisation and implementation of BDI agents constitutes the research of many scientists [4], [12], [15], [16], [20], [23]. Some of them criticise the necessity of studying multi-modal logic for the formalisation and construction of such agents, because they have not been completely axiomatised and they are not computationally efficient. Rao and Georgeff [21] state that the problem lies in the big distance between the powerful logic for BDI systems and practical systems. Another problem is that this type of agent is not able to learn, a necessary requirement for them since they have to be constantly adding, modifying or eliminating beliefs, desires and intentions. It would be convenient to have a reasoning mechanism which would enable the agent to learn and adapt in real time i.e. as the computer program is executing rather than have to recompile such an agent whenever the environment changes.

It has been demonstrated that BDI agent implemented using a case-based reasoning system can substantially solve the two problems previously mentioned [8], [14]. Implementing agents in the form of CBR systems also facilitates their learning and adaptation. Among the different disciplines of cognitive science, cognitive psychology has widely shown the importance of learning from experience [3]. If the

proper correspondence between the three mental attitudes of BDI agents and the information manipulated by a case-based reasoning system is established, an agent with beliefs, desires, intentions and a learning capacity will be obtained. Although the relationship between agents and CBR systems has been investigated by other researchers [1], [10], [19], [26], we have proposed an approach, whose main characteristic is its direct mapping between the agent conceptualisation and its implementation, in the form of a CBR system [8], [13], [14]. Intelligent and autonomous agents may then be implemented for working on the Internet automatically, in an unsupervised way, and then may be capable of evolving without supervision. The concept of case based reasoning system is presented and then it is shown how this methodology may be used to construct autonomous agents.

2.1 Case-based Reasoning Systems

Case-based reasoning (CBR) is used to solve new problems by adapting solutions that were used to solve previous similar problems [22]. To reach this objective, CBRs are based on the knowledge stored in their memory, in the form of cases or problems. Figure 1 shows the reasoning cycle of a typical CBR system that includes four steps that are cyclically carried out in a sequenced way: retrieve, reuse, revise, and retain [24]. Each of the reasoning steps of a CBR system can be automated, which implies that the whole reasoning process could be automated to a certain extent [7], [11]. This assumption means in our opinion, that agents implemented using CBR systems could be able to reason autonomously and therefore to adapt themselves to environmental changes. The automation capabilities of CBR systems have led us to establish a relationship among cases, the CBR life cycle, and the mental attitudes of BDI agents. Based on this idea, a model that facilitates the implementation of the deliberative BDI agents (for the Internet) using the reasoning cycle of a CBR system is presented.

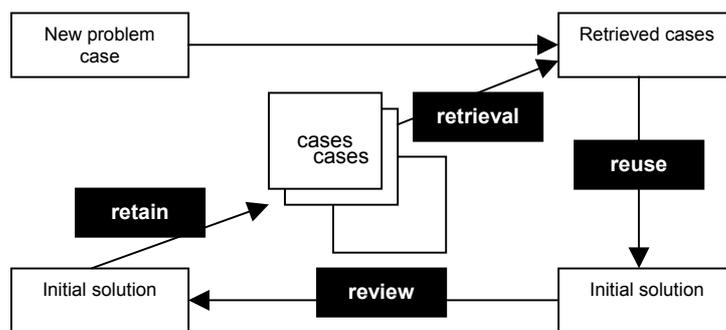


Fig. 1. CBR Cycle of Life

2.2 Automating the construction of agents using CBR systems

This section identifies the relationships established between deliberative BDI agents and CBR systems, and shows how an agent can reason with the help of a case-based

reasoning system. Our proposal defines a direct mapping from the concept of an agent to the reasoning model, paying special attention to two elements: (i) how the mapping should allow a direct and straightforward implementation of the agent and (ii) how the agent is able to learn and evolve with the environmental changes. Properties that may be very useful to dynamic Internet applications. In the presented model, the CBR system is completely integrated in the agents' architecture, referring to the above-mentioned works, in which the agents see the CBR system just as a reasoning tool. Our proposal is also concerned with the agent's implementation and presents a "formalism" easy to implement, in which the reasoning process is based on the concept of intention. In this model, intentions are cases, which have to be retrieved, reused, revised and retained [8], [13].

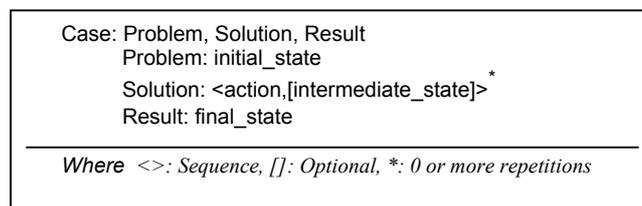


Fig. 2. Definition of a case in a case-based reasoning system

To achieve both goals, the structure of the CBR system has been designed around the concept of a case. The problem, the solution and the result obtained when the proposed solution is applied make a case. Figure 2 shows these components: the problem defines the situation of the environment at a given moment, the solution is the set of states undergone by the environment as a consequence of the actions that have been carried out inside it, and the result shows the situation of the environment once the problem has been solved.

Figure 3 defines what are the beliefs, desires and intentions for a BDI agent. Each state is considered a belief; the objective to reach may also be a belief. The intentions are plans of action that the agent is forced to carry out in order to achieve its objectives [2], so an intention is an ordered set of actions; each change from state to state is made after carrying out an action (the agent remembers the action carried out in the past when it was in a specified state, and the subsequent result). A desire will be any of the final states reached in the past (if the agent has to deal with a situation, which is similar to a past one, it will try to achieve a similar result to the previously obtained result).

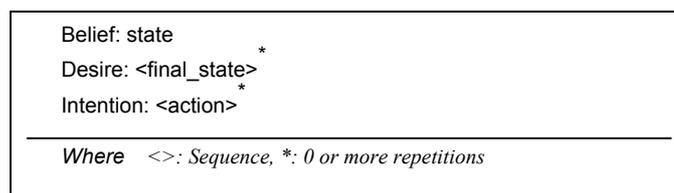


Fig. 3. Definition of the mental attitudes of a BDI agent

The relationship between CBR systems and BDI agents can be established implementing cases as beliefs, intentions and desires which caused the resolution of the problem. The obvious relationship between BDI agents and CBR systems can be identified by comparing Figures 2 and 3.

Using this relationship we can implement agents (conceptual level) using CBR systems (implementation level). Then we are mapping agents into CBR systems. The advantage of this approach is that a problem can be easily conceptualised in terms of agents and then implemented in the form of a CBR system. So once the beliefs, desires and intentions of an agent are identified, they can be mapped into a CBR system. The case based reasoning system may use then a Variational Calculus based planner (VCBP) to identify its actions plans in execution time using past experiences and the user feedback [13]. Variational calculus can be used to obtain the most adequate plan to achieve a goal in environment with uncertainty, during the CBR retrieval and reuse stage. The case-tool VCBP-Agents automates the implementation of such Internet agent based systems and the specifications of the believes, desires and intentions of the agents [14].

3 A CBR-BDI based engineering sales support system

The construction industry is an information intensive economic sector. This activity, as many others, require the use of a great amount of data, ranging from product data to technical publications, from buildings regulations to best practice guides. This section describes an information system that has been developed for a construction company, D&B Constructions, in which the previously mentioned agents have been used. This distributed agent based system helps the company to make as much profit as possible from the information published on the Internet and the information that the company holds, and to reuse it as much as possible especially to estimate budgets.

The e-business engineering sales support system incorporates several specialised agents that search for and organise information and data, and several assistant sales support agents. The multiagent system has been implemented in Java using servlets (running in an Apache, Tomcat, jserv and Linux environment). The specialised agents are Java applications that run on the company Intranet and the assistant agents run in a portable computer connected to the Internet via a mobile phone.

The D&B Constructions deals with medium to small construction problems and it specialises in installing heating and air conditioning systems in a wide area of the Northwest of Spain. They have a sales force that is growing continuously, which implies that continuously new salesmen are taken on board without much experience in many cases. Until now the salesmen had to visit the clients on demand, had to take notes of their problems and then they had to contact an engineer or an experienced salesman, which had to estimate the work price and personnel and material required to carry on the work. The system here outlined was developed to aid the sales force, and in particular the inexperienced personnel, in the estimation of costs, thus reducing the process bureaucracy.

3.1 The System Architecture

In the expansion policy of B&D Constructions one of the main points is its incorporation of new technologies. Several steps will be taken in this direction for developing a web based information system that allows the company to publish information about their activities and that facilitates the communication between the administration, the sales force, the providers and the clients.

Figure 4 presents the architecture of the multiagent system. The planning agent has been implemented with the architecture described in [8], [14]. It estimates the construction cost, and the personnel and material required to carry out a construction project. It also generates reports about clients (or potential clients) using the information stored in the company databases and the one obtained by the Web search agent from the web. The planning agent generates working plans using their incorporated CBR system. The Web agent incorporates a web search engine that looks continuously for potential clients, information about them, new providers and products. This agent starts looking from a predetermined web address and searches for new ones using natural language processing strategies, optimised for the Web, based on the combination of the grammatical characterisation of words and statistical decision techniques [5]. This agent is monitored and guided by a marketing expert. Assistant agents (they can be as many agents as salesmen) are interface agents that facilitate the communication between the salesmen and the planning agent, they also hold summary information about the clients visited by its salesman owner and by the rest of the salesmen.

Before a salesman visits a client, he/she interrogates his/her assistant agent providing a description of the client (Name, Address and Activity). The assistant agent compares this data with previous queries and stored knowledge, using the VCBP incorporated by the CBR system, to identify the most suitable installation. This information is related to previous building work carried out for the client, his financial status, comments about him, noted by the firm's personnel during previous relations with this client, location information and other potentially useful data. This information is valuable especially when an inexperienced salesman starts a negotiation process. If the assistant agent cannot help the salesman or if the salesman demands more information, his assistant agent contacts the planning agent, which searches for information about the client in its case-base. This agent also interrogates the Web search agent asking for information about clients. The Web search agent obtains information from the web, analyses and indexes it using natural language processing algorithm optimised for Internet, as mentioned above. Information about potential clients, new materials and providers is sent to the administration agent, which can be interrogated by any of the Construction Company managers, engineers or sales supervisors. They can, then, use this pruned information to target new business. The administration agent is an interface agent that facilitates the interaction between the users (Company managers) and the rest of the elements of the system: agents, databases and even salesmen.

As mentioned above, the multiagent [25] system has been built using a Java-based library. This library is an extension of the one used to implement the STEB (Simulated Tactical Environmental Bubble) system [6], [7]. The STEB system was designed to forecast the temperature of ocean waters ahead of ongoing vessels. It is a

multiagent system composed of several software agents that co-operate between them and that are operational in different locations: war/oceanographic vessels and in an Oceanographic Laboratory (Plymouth Marine Laboratory). The agents installed on the vessels use hybrid Case Based Reasoning-Artificial Neural Network (CBR-ANN) system [1] to forecast and communicate with the rest of the agents using Knowledge Query and Manipulation Language (KQML) performatives.

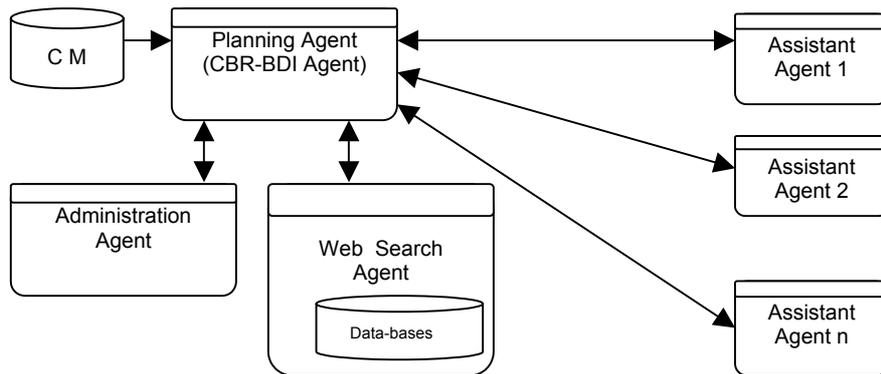


Fig. 4. Agent-oriented architecture

Similarly to the STEB system, when constructing the e-business multiagent system here presented, a decentralised architecture was selected, in which agents interact between one another when they require information or need to share data. The agents communicate with each other using a message passing protocol. Such messages are KQML performatives. The agents of this system collaborate between each other sharing information and working together to achieve a given goal. They use a simple collaboration mechanism. For example, if Salesman A is associated with the Assistant Agent A, and visits a Client X, the Assistant Agent A has to contact (send the problem of the Client X, via a performative) the Planning Agent. Then the Planning Agent generates the solution plan, and sends it back to the Assistant Agent [14].

In a system of these characteristics, data security has to be taken into consideration. A Role-based Access Control with elements that allow the certification of operations has been implemented to guarantee the data security and the information protection [9]. This security system protects the knowledge-base and the information stored in the system from external “agents” or non accredited personnel.

4 Results and Conclusions

The acceptance of this agent-based distributed system, by the Company staff, has been excellent. The system has been used in 1450 potential sales. D&B Constructors hold a data base with over 7810 installations from January 1997 till now.

In 92% of the system interrogations the estimation of the CBR-BDI agent differed by less than 3% of the one given by an expert salesman, and in the 6% of the enquires, it differed by less than 7%. Only in 2% of the occasions the agent output differed in more than 7% to the expert estimation. These deviations were caused by two reasons: the client required a combination of installations and equipment, which did not appear in any of the cases stored in the agent case-base, or there were errors in the system communication or in the data collection. These errors may be minimised during the review phase. With respect to this point, strategies are under investigation to detect such error and to identify problems and/or “generated plans” with potential risks.

It is expected that the accuracy of the business solution will increase as more cases are introduced in the planning agent memory. Company experts have estimated that the use of this agent-based system could reduce the installation sales cost up to 42% of the actual cost, and the time of the sale up to 45%.

The agent-based architecture presented in this paper facilitates the construction of dynamic Internet based systems. The reasoning capacities of the agents help them to solve problems, facilitate its adaptation to changes in the environment and to identify new possible solutions. New cases are continuously introduced and older ones are eliminated. Mora *et al.* [18] have described the gap that exists between the formalisation and the implementation of agents. What we propose in this article is to define the beliefs and intentions clearly (with a high level of abstraction), and to use them in the life cycle of the CBR system, to obtain a direct implementation of an autonomous agent. This paper has shown how agents can be developed with this technology, and how such agents can be successfully used to construct autonomous agent based system for the Internet.

Acknowledgement

This work has been partially supported by the CICYT projects TEL99-0335-C04-03 and SEC2000-0249, the PGIDT00 project MAR30104PR and the project SA039/02 of the JCyL.

References

1. Bergmann R and Wilke W. (1998) Towards a New Formal Model of Transformational Adaptation in Case-Based Reasoning. European Conference on Artificial Intelligence (ECAI'98), John Wiley and Sons.
2. Bratman M.E., Israel D., and Pollack M.E. (1988) Plans and resource-bounded practical reasoning. *Computational Intelligence*, 4, pages 349-355.
3. Caplan L.J. and Schooler C. (1990) Problem Solving by Reference to Rules or Previous Episodes: The Effects of Organized Training, Analogical Models, and Subsequent Complexity of Experience. *Memory & Cognition*, 18(2), pages 215-227.
4. Cohen P.R. and Levesque H.J. (1990) Intention is choice with commitment. *Artificial Intelligence*, 42(3).
5. Corchado J. M. (2001) CBR Agents in an e-commerce environment. Proceedings of WOOPS-ECOOP, Budapest, June.

6. Corchado J. M. and Aiken J. (2003) Hybrid Artificial Intelligence Methods in Oceanographic Forecasting Models. *IEEE SMC Transactions Part C*. Vol.32, No.4, November 2002: 307-313.
7. Corchado J. M. and Lees B. (2001) A Hybrid Case-based Model for Forecasting. *Applied Artificial Intelligence*. Vol 15, no. 2, pp105-127.
8. Corchado J. M., Laza R., Borrajo L., Yañes J. C. and Valiño M. (2003) Increasing the Autonomy of Deliberative Agents with a Case-Based Reasoning System. *International Journal of Computational Intelligence and Applications*. In press. ISSN: 1469-0268.
9. Corchado J.M., Aiken J, Rees N. (2001) Artificial Intelligence Models for Oceanographic Forecasting. Plymouth Marine Laboratory. ISBN-0-9519618-4-5.
10. Feret M. P. and Glasgow J. I. (1994) Explanation-Aided Diagnosis for Complex Devices, *Proceedings of the 12th National Conference an Artificial Intelligence, (AAAI-94)*, Seattle, USA, August 94.
11. Fyfe C. and Corchado J. M. (2001) Automating the construction of CBR Systems using Kernel Methods. *International Journal of Intelligent Systems*. Vol 16, No. 4, April 2001. ISSN 0884-8173.
12. Georgeff M. P. and Lansky A.L. (1986) Procedural knowledge. In *Proceedings of the IEEE Special Issue on Knowledge Representation*, volume 74. pages 1383-1398.
13. Glez-Bedia M. and Corchado J. M. (2003) Constructing autonomous distributed systems using CBR-BDI agents. *Innovation in knowledge Engineering*. Faucher C., Jain L. and Ichalkaranje N. (eds.). Springer-Verlag. In press.
14. Glez-Bedia M, Corchado J. M., Corchado E. S. and Fyfe C. (2002) Analytical Model for Constructing Deliberative Agents, *Engineering Intelligent Systems*. Vol 3: 173-185.
15. Jennings N.R. (1992) On Being Responsible. In Y. Demazeau and E. Werner, editors, *Decentralized A.I. 3*. North Holland, Amsterdam, The Netherlands.
16. Kinny D. and Georgeff M. (1991) Commitment and effectiveness of situated agents. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI'91)*, pages 82-88, Sydney, Australia.
17. Kinny D., Ljungberg M., Rao A.S., Sonenberg E.A., Tidhar G, and Werner E. (1994) Planned team activity. In *Artificial Social Systems. Lecture Notes in Artificial Intelligence (LNAI-830)*. Amsterdam, Netherlands. Springer Verlag.
18. Mora M. C., Lopes J. G., Viccari R. M. and Coelho H., (1998) BDI Models and Systems: Reducing the Gap. *ATAL-98*.
19. Olivia C., Chang C. F., Enguix C.F. and Ghose A.K. (1999) Case-Based BDI Agents: An Effective Approach for Intelligent Search on the World Wide Web", *AAAI Spring Symposium on Intelligent Agents*, 22-24 March 1999, Stanford University, USA
20. Rao A. S. and Georgeff M. P. (1991) Modeling rational agents within a BDI-architecture. In J. Allen, R. Fikes, and E. Sandewall, editors, *Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning*. Morgan Kaufmann Publishers, San Mateo, CA.
21. Rao A. S. and Georgeff M. P. (1995) BDI Agents: From Theory to Practice. *First International Conference on Multi-Agent Systems (ICMAS-95)*. San Francisco, USA, June.
22. Riesbeck, C. K. and Schank, D. B. (1989) *Inside Case Based Reasoning*. Erlbaum.
23. Shoham Y. (1993) Agent-Oriented programming. *Artificial Intelligence*, 60(1): pages 51-92.
24. Watson I. and Gadingen D. (1999) A Distributed Case-Based Reasoning Application for Engineering Sales Support. In *Proceedings of IJCAI-99*. Morgan Kaufmann Publishers Inc. Vol 1, pp 600-605.
25. Weiß G. (1996). Adaptation and learning in multi-agent systems: some remarks and Bibliography. *Proceedings IJCAI'95 workshop*. Montreal, Canada. August, 1995.
26. Wooldridge M. (1999) Intelligent Agents. *Multiagent Systems. A modern approach to Distributed Artificial Inteligence*. Edited by Gerhard Weiss. Pages 27-77.

27. Wooldridge M. and Jennings N. R. (1994) Agent Theories, Architectures, and Languages: A Survey. Procs. ECAI-94 Workshop on Agent Theories, Architectures, and Languages.