Modelling Intelligent Agents and Assistants in a Call Centre Environment

Guy Saward
University of Hertfordshire
College Lane, Hatfield,
Herts AL10 9AB, UK
Phone: +44 (0) 1707-284375
Email: g.r.saward@herts.ac.uk

ABSTRACT: The call centre environment is a fertile area for the application of intelligent software agents. This paper examines a number of problems that can be addressed through the application of agents built using a knowledge-based approach. Case-based reasoning is used as the basic knowledge representation within a 3-tier architecture. The use of cases as a knowledge representation technique for agents within a call centre environment over comes some of the issues typically associated with knowledge-based agents.

KEYWORDS: case-based reasoning, agents, architecture, call centres

INTRODUCTION

The call centre can be a rich environment in terms of the number of business processes and decisions that need to be made, the amount of knowledge that is required to support those processes, and the number of systems that need to be accessed to process any decision. This is particularly true with the convergence of call centres, help desks and latterly the Internet. Call centres have traditionally been seen as having high volumes of low complexity transactions while help desks have focused on smaller volumes of more complex issues. The convergence of these in the “contact centre” has resulted in large-scale operations needing to handle high volumes of calls across a wide range of complexities.

The use of intelligent agents and assistants provides an opportunity to increase service volumes while maintaining quality and value across all calls. The addition of Internet access allows contact centres to deploy existing knowledge assets to allow self-service by customers. We shall look at both the definition of intelligent agents and the key issues in Call Centres before reviewing the decision making areas where intelligent agents could be used. We describe the general capabilities of case base reasoning before identifying specifically how it can be used to capture the knowledge required of intelligent agents and assistants.

DEFINING AGENTS

There are a plethora of definitions of what is and is not a software agent, the number of which continues to grow from those presented by Franklin and Graesser (1996). Definitions range from the philosophical to the behavioural. Shoham (1997) does the former in defining an agent as "… an entity whose state is viewed as consisting of mental components such as beliefs, capabilities, choices and commitments”. Nwana (1996) is one exponent of a behavioural approach by requiring agents to exhibit at least two of the following: autonomy, co-operation and learning.

For our purposes the key recurring behavioural characteristic is that of agency, defined by Gilbert (1995) as the "degree of autonomy and authority vested in the agent". Agency underpins an agent's decision-making capabilities that enable it to undertake some specific task. Varying degrees of agency can be seen in a functional characterisation that looks at typical uses of agents identified by Maes (1994).

The tasks identified by Maes can be ranked by increasing levels of authority and capability and range across:

- monitoring events and procedures through the application of domain knowledge;
- training the user through the presentation of information and recommendation of actions;
- hiding the complexity of difficult tasks and removing some decision making from users, for example those involved in collaborative activities such as meeting scheduling
- performing tasks on behalf of the user in which the user may define the parameters, goals or tasks but the agent has full authority in acting on its domain knowledge.

All of these tasks can be found within the call centre and is indicative of the applicability of agent technology to the call centre environment.

CALL CENTRE DECISION MAKING

Industry experience (Kjellerup, 1998) highlights two key issues for successfully implementing and maintaining a call centre :

- exceeding customer's service expectation;
- reducing the time and cost of the customer contact process over time.

Factors in achieving this include identification of best practice, consistency in call handling, multi-skilling agents, and improving capacity through reducing call times. In parallel with cost reduction, service providers are looking to utilise call centres as revenue generating business units (e.g. Allimadi 1999). This trend will be increased as call deflection and self-service through the use of the Internet resources and interactive voice response systems reduce the call volumes (Hibbard. et al, 1998)

Key decisions in call handling include:

- **categorisation:** what is the primary purpose of the call? Is it to enquire about prices, trouble shoot a technical problem or to pay a bill.
- **routing:** can the call be resolved using an automated service such as fax back, or failing that, which agent within the call centre has the technical, personal and language skills to deal with the call?
- **problem identification:** what is the exact nature of the request and/or the root cause of any problem?
- **problem resolution:** what actions need to be taken to resolve the call?
- **presentation of information:** how should sensitive information be requested or presented? For sales calls this may cover how objections should be handled, or what phrasing should be used for regulatory compliance.

Each of types of decision making identified above requires particular organisation, product and procedural knowledge. If these functions are to be taken over by software agents, then this knowledge must be captured and made accessible to the agent.

KNOWLEDGE-BASED AGENTS

Using the characterisation of agents as presented by Gilbert et al (1995), decision making agents in the call centre need to focus on the dimension of intelligence. This is one of the two key issues that Maes (1994) identified when she focussed on the need to consider competency and trust when building software agents. The issue of competency is concerned with imbuing the agent with sufficient knowledge to act effectively while the issue of trust revolves around getting the user to vest the appropriate authority in the agent to allow it to act on those decisions.

Within the call centre environment it is useful to make a distinction between agents that take business decisions and act on behalf of the user and the notion of an assistant which supports a user in the execution of a specific task. In this paper we reserve the term agent for those applications which demonstrate the highest level of agency and exclude interface and collaborative agents in which:

- decision making control rests with the user;
- the agent's task focus is on the selection and presentation of information;
- the agent mediates the user's relationship with their systems and information environment but does not control it.

Given the distinction between agents and assistants it is clear that many of the applications that have been built to support event monitoring, training and decision making in the call centre should be viewed more as assistants than agents. In order to increase the agency of call centre applications, decision-making knowledge needs to be packaged in a way that promotes autonomy and increases agency. In doing this we need to address specific problems related to competency and trust that Maes raises when trying to create agents using a knowledge-based approach.

First, agents require a detailed knowledge base to be competent in a specific task. The investment in a sufficiently rich knowledge base would be prohibitive given the task specific nature of the resulting agent. Second, the knowledge captured would not be adaptable and would result in an agent that is limited in its competency. Third, a pre-programmed complex knowledge base would not be explicable to the user and any autonomous agent whose decisions were based on it would not be given the appropriate authority. In the rest of this paper we go on to describe how case-based reasoning (CBR) can be used to model agent's decision making processes that addresses these problems.

THE PRINCIPLES OF CASE-BASED REASONING

For CBR systems, as described by Watson (1997), a *case* is the basic unit of knowledge and provides a description of a particular situation and the action that is appropriate in that situation. The features that uniquely characterise a situation are described as a set of attribute-value pairs or more formally as a tuple in an n-dimensional space. Similar attribute value pairs can be used to characterise a problem situation and are used to identify the closest matching solution case. The archetypal reasoning cycle used by a CBR system, defined by Kolodner (1993), involves:

- presenting a set of problem features;
- matching the problem to known cases;
- adapting the nearest matching case if required;
- applying the action from the closest matching case;
- updating the knowledge base as required with a new case summarising the problem features and the result of the action.

By way of an example, consider the agents within an IT call centre agent (human or system) that may be used to resolve in-bound calls. Each agent, be it human or software, may have a number of different profiles describing the types of calls that could be routed to it. The profile attributes might include: languages that the agent can converse in; level of product skills in specific applications, operating systems and hardware platforms; and functional skills such as sales, billing or support. A call routing agent would o-operatively identify the skills required for a particular call and match that profile against all available agents to identify the most appropriate agent to take the call. If no perfect match were found then the best match would be used. If a successful resolution were achieved then the agent profile could be updated to take account of the successful resolution.

Simple case bases as described above can be used to implement simple business rules associated with call classification or call routing. More complex, iterative case base searches can be used to implement co-operative problem solving or process execution. In this search model, problem features are not all presented simultaneously but are investigated using an iterative case matching process. For example, if a PC user has problems booting their PC a diagnostic agent might first ascertain whether the PC can be booted from floppy or in protected mode before suggesting further tests and possible solutions.

A CBR AGENT ARCHITECTURE

The use of CBR technology for assisting human agents in Call Centres is now a routine occurrence. Applications range from PC and mobile phone trouble shooting, through service delivery such as insurance claims validation and utility billing, to advice on tax, building and planning control and social security benefits. However, the majority of these applications would be classified as assistants rather than as agents as they would fail to satisfy an autonomy criterion – they do not run continuously, asynchronously or unsupervised. They would also be seen to be weak on the co-operative criterion as they are typically only co-operating with a single, human agent.

The author has been involved in a number of more sophisticated CBR applications have been developed and deployed that do meet the criteria for intelligent agents. They have been based on a three tier architecture, as described for a particular application by Saward (1999).

The levels in the case-based agent architecture are shown in figure 1 and include:

- an agent wrapper, that enables the agent to be embedded in a particular system environment;
- a meta-code interpreter that is used to code interactions with other agents.
- a case base search engine that determines the sequence of actions of the agent;

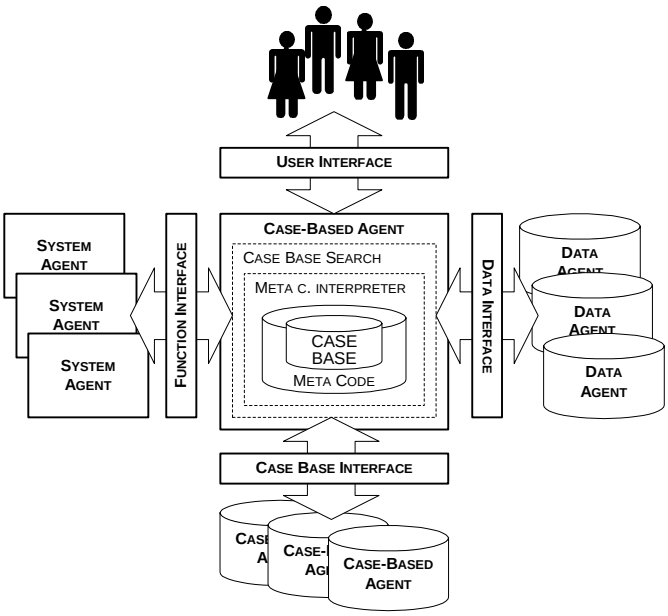

Figure 1: The Case-Based Agent Architecture

The four interfaces identified allow the case-based agent to request information or action from other systems. These systems could be fully fledged agents, sub-agents or simple applications. The distinction between the different types of interfaces could be considered artificial but is used to highlight the potential differences in agency in the supporting systems. It is also useful to distinguish between requests for data that return information which is filtered and acted on by the case-based agent, and function requests which may be made asynchronously and whose results are not of interest to the case-based agent.

This architecture allows for a CBR agent to be deployed within a desktop front-end environment, or as a backend server or Internet process. In one particular application, this architecture has been instantiated with Inference's CBR3 case base search engine and an OLE object wrapper around a meta-code interpreter written in VB5. The agent was deployed both on the desktop within a 200 seat call centre and as a stand alone process taking input from an email server. Only one very small change to the business process knowledge was required. An input validation check was included to ensure that sufficient key data was included before detailed processing was started.

CONCLUSIONS

One of the principle claims made for CBR (by Kolodner and Watson amongst others) is that it represents a more intuitive mode of reasoning that degrades more gracefully than the use of rules typically used in "traditional" expert systems. Within the customer service market this is born out by the dominance of case-base reasoning tools within the problem resolution market. What is yet to be seen is whether the use of a high level knowledge representation is sufficient advantageous to merit its wide spread inclusion in agents developed in more typical agent programming languages such as Telescript, Java, Agent Tcl and Obliq (as identified by Kiniry and Zimmerman (1997). However, what is clear is that the use of CBR in modelling agents within the call centre environment negates the objections raised by Maes that were previously identified.

In general, using a case-based reasoning paradigm results in a knowledge base that is easily accessible to end users. Each case is a self contained script or scenario that includes all the reasoning steps for a particular situation. For complex processes, graphical notations can be used to identify and structure cases and show the relationships between cases (see Saward 1999 for examples). The accessibility of the knowledge in the case base facilitates its change and update and means that it can be readily adapted or customised for particular situations.

There are additional specific features of the call centre market that make the production of competent agents possible. First, many organisations are already capturing business knowledge in formats that can easily be modelled as cases. This knowledge capture is a typical part of the introduction of call centres when an organisation identifies the best business practice that it would like to see consistently applied within the call centre. Second, in some situations the primary purpose of the call centre cannot actually be achieved without systematic codification and access to this knowledge and this investment is a key part of the cost/benefit business case. In future, it is possible that the high cost of such investment may be amortised across a larger user base as same knowledge is delivered through alternative lower cost channels such as CD-ROMs and the Internet.

The actual process of building the cases is a key part in generating the trust that users need to have in an agent. Typical user-centered development methodologies mean that the knowledge built into the case base often comes from key, experienced users who are influential in promoting the system. In addition, the accessibility of the knowledge means that what has actually been built can be easily verified and understood. The inclusion of user testing will also promote the required trust. Finally, it is worth considering that within the call centre and the applications identified above there is not the individual relationship between a user and an agent that occurs with other agents such as user interface, inter or mail filtering agent. The "user" of the agent is the call centre organisation itself and their trust should be apparent in the final system sign-off.

REFERENCES

Allimadi, M., 1999. Banking and Financial Institutions' Call Centers Eye Big Bucks in *Call Centre Magazine*, February 1999

Franklin, S., and Graesser, A., 1996. Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents, in *Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages*, Springer-Verlag

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.

Hibbard, J., Dalton G., Thyfault, M., 1998. Web-Based Customer Care -- Businesses turn to the Internet to improve customer service and lower costs, in *InformationWeek*, Issue: 684, June 01, 1998.

Kiniry, J. and Zimmerman, D. 1997. A Hands-On Look at Java Mobile Agents, in *IEEE Internet Computing*, Vol. 1(4)

Kjellerup, N., 1998. Why Call Centres Fail, Australian Telecomms User Group Conference, Melbourne, http://www.callcentres.com.au/why.htm.

Kolodner, J. 1993. Case-Based Reasoning. San Mateo, CA: Morgan Kaufmann.

Maes, P., 1994. Agents that Reduce Work and Information Overload, in *Communications of the ACM*, Vol. 37(7)

Nwana , H, 1996. Software Agents: An Overview, in *Knowledge Engineering Review*, Vol. 11(3)

Saward, G. 1999. Using Cases for Process Modelling: An Example from the Water Supply Industry, in *Proceedings of the 12th International Conference on Industrial and Engineering Applications of AI and Expert Systems*, Springer-Verlag forthcoming

Shoham, Y, 1997. An Overview of Agent-Oriented Programming, in Bradshaw, J., *Software Agents*, MIT Press

Watson, I. 1997, Applying Case-Based Reasoning: Techniques for Enterprise Systems. Morgan Kaufmann