AMBIENCE & COLLABORATION
Embedded Agents in a Human-centered World

M. J. O'Grady, G. M. P. O'Hare, R. Tynan
CLARITY- Centre for Sensor Web Technologies, School of Computer Science & Informatics, University College Dublin
Belfield, Dublin 4, Ireland

R. W. Collier, C. Muldoon
School of Computer Science & Informatics, University College Dublin, Belfield, Dublin 4, Ireland

Keywords: Ambient Intelligence, Embedded agents, Human-centered computing.

Abstract: Supporting people in the pursuit of their everyday activities is a laudable objective and one which researchers in various disciplines including computing, actively seek to accomplish. The dynamic nature of the end-user community, the environments in which they operate, and the multiplicity of tasks in which they engage in, all seem to conspire against the desired objective of providing services to the end-user community in a transparent, intuitive and context-aware fashion. Indeed, this inherent complexity raises fundamental problems for software engineers as they frequently lack the tools to effectively model the various scenarios that dynamic user behaviour give rise to. This difficulty is not limited to exotic applications or services; rather, it is characteristic of situations where a number of factors must be identified, interpreted, and reconciled such that an accurate model of the prevailing situation at a given moment in time can be constructed. Only in this way, can services be delivered that take into account the prevailing human, social, environmental and technological conditions. Constructing such services calls for a software solution that exhibits, amongst others, diffusion, autonomy, cooperation and intelligence. In this paper, the potential of embedded agents for realising such solutions is explored.

1 INTRODUCTION

Computing is a multi-faceted concept and is perceived in a number of diverse ways. Quite how it is perceived is influenced by a number of factors but one prominent issue concerns the role and prominence that computing plays in a persons' everyday life. For some, designing, implementing and administrating computing systems is their everyday task. For others, computing forms an indispensable tool in the fulfilment of their work. However, for a significant number, software applications and services have a peripheral role in their lives. Why this is the case must remain a matter of conjecture. However, it does indicate that a significant opportunity exist for harnessing computational resources for the benefit of people in the pursuit of their everyday lives – Ambient Assisted Living (Nehmer at al, 2006) being a particular case in point. To avail of this opportunity it beholds software professionals to demonstrate clear and tangible benefits if their technological solutions are to be adopted. Financial considerations, though important, are not paramount. Moreover, the learning curve must be short and the services themselves easy and intuitive to use. Achieving these objectives is, of course, a challenging and formidable task.

In focusing on the technological, there is a significant risk that the desires and requirements of people may be neglected. For example, people may be perceived as black-boxes - objects about which very little is known about, and the circumstances in which they operate inscrutable. Alternatively, people may be treated as a homogenous entity and ascribed a worldview remarkably similar to the designers of a piece of hardware or software. In either case, people lose. Clearly, focusing solely on the technology is a deficient approach, and to remedy this, it is necessary to obtain a broader picture of the

J. O'Grady M., M.P. O'Hare G., Tynan R., W. Collier R. and Muldoon C. (2009).
AMBIENCE & COLLABORATION - Embedded Agents in a Human-centered World.
In *Proceedings of the 4th International Conference on Software and Data Technologies*, pages 361-364
DOI: 10.5220/0002281503610364

prevailing situation at the time people are actively using applications and services or, in other words, their context (Greenberg, 2001).

2 A QUESTION OF CONTEXT

Before a context can be incorporated into the design of a service, it is necessary to articulate the elements of context that the services in question require. It must then be ascertained how the necessary contextual elements can be captured. In some cases, this may be accomplished by asking the user in question. In other cases, the application may be able to dynamically determine the device parameters under which it is operating and adapt accordingly, displaying some limited autonomic behaviour in the process (Kephart and Chess, 2003). Finally, individual contextual elements may be determined using suitably equipped sensors and Wireless Sensor Networks (WSNs). Assuming that the required contextual elements are available, software designers may need to consider if the cost, both computationally and financially, of capturing context is worth the effort. Of particular relevance to this discussion is the case where individual contextual elements must be explicitly and continuously monitored such that a model of user behaviour may constructed that would enable the identification of scenarios where proactive intervention would aid the user in the accomplishment of the task at hand. Such an intervention would only be possible after a significant quantity of context-related data had been captured, filtered and characterised. Identifying patterns in individual context data streams, and reconciling those with patterns identified from other context data streams, as well as patterns from previous behaviour models, is a computationally-intensive task and suggests the need for harnessing intelligent techniques.

3 STRATEGIES FOR REALISING INTELLIGENCE

An array of Artificial Intelligence (AI) techniques exists, and depending on the requirements of the services in question, any individual or combination of techniques may be harnessed as appropriate. However, the use of such techniques incurs a cost which may be measured in at least two ways. Firstly, AI techniques are notoriously computationally intensive. Thus the immediate availability of powerful workstations is a prerequisite. However, if an interconnecting network that incorporate a wireless component is utilised, there may be problems due to latency and poor data throughput. This leads directly to the second cost penalty which concerns the perceived response time and, ultimately, the usability of the service in question. Recall that the primary objective is to incorporate proactive and anticipatory behaviour for aiding users in the fulfilment of their everyday tasks. A small window of opportunity exists where such behaviour can be fruitfully activated. Hence, the response time to changes in the various contextual elements must be such that these opportunities can be availed of in a timely fashion.

It is interesting to reflect at this juncture that the Ambient Intelligence (AmI) (Vasilakos and Pedrycz, 2006) initiative, as its name suggests, envisages the incorporation of intelligent techniques. To recall: AmI was formulated in response to anticipated difficulties arising in scenarios where ubiquitous computing environments were deployed. Such environments would incorporate a significant number of artefacts or objects that had been augmented with computational technology. As such artefacts increasingly proliferate, a scenario was envisaged arising where competition for the user's attention would be at a premium. The net result being that such environments would be unusable and inhospitable to users - a most unfortunate state of affairs when the original objectives are considered. Thus AmI anticipates the use of intelligent techniques, especially Intelligent User Interfaces (Maybury and Wahlster, 1998), as a means of managing interaction within the environment and minimising the need for explicit intervention. For the purposes of this discussion, we are interested in the potential of AI for identifying opportunities where the user may be opportunistically aided in the fulfilment of their tasks. AmI complements this objective but does not seek to fulfil it.

To develop and maintain an accurate model of user behaviour, it is essential to harness as much contextual data as possible, using an array of sensors and other devices. Ideally, these data streams would be processed as near to their source as possible, implying the need to employ a Distributed Artificial Intelligence (DAI) or agent-based approach. In the later case, one practical implementation for resource-bounded devices is that of embedded agents.

3.1 Embedded Agents

Historically, agents have been deployed on fixed networked nodes where access to computational resources is not a problem. However, developments in mobile and embedded technologies, both hardware and software related, are extending the reach of agent technologies towards the extreme periphery of the network. Indeed, examples of embedded agents for AmI applications have been documented (Keegan et al, 2008; O'Grady et al, 2005; Hagras et al, 2004). The current range of PDAs and smartphones are capable of running multi-tasking and multi-threaded applications. Admittedly, the screen size of ¼ VGA is one obvious limitation but it is a significant improvement over the screens that supported a few lines of text that were state-of-the art just a few short years ago. In addition, a number of popular communications protocols are supported. However, it is developments in tools and languages for designing and developing software for mobile devices that has had the most impact. In particular, a significant number of mobile devices comprise a Java Virtual Machine (JVM) compliant with the Java ME specification.

Embedding agents (Figure 1) on sensor devices may appear impractical initially but recent developments suggest otherwise. A number of sophisticated sensor platforms are commercially available whose specifications are similar to many PDAs and mobile phones, albeit not from a GUI perspective. A JVM is available for such platforms thus enabling them to support relatively sophisticated applications, either in isolation or, more likely, as a node in a distributed application or service. In a standard Wireless Sensor Network (WSN) topology, sophisticated sensors could act as base stations, collecting data from nearby leaf nodes, sharing it with other nodes and collaborating with them to make sense of the data.

From a software platform perspective, a number of embedded agent platforms have been developed in the past number of years. O'Hare et al (2006) provides a useful survey of some common platforms. One particular example of a system that demonstrates the feasibility of agents on WSNs is Agilla (Kok et al, 2005). This is a middleware solution that adopts mobile agents for coordination and migration in the fulfilment of WSN specific tasks.

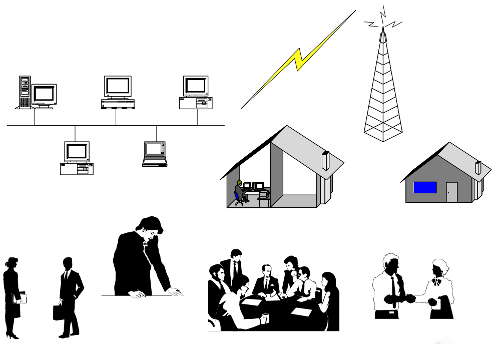

Figure 1: Embedded agents can be deployed on suitably equipped devices ranging from workstations to smart phones to embedded sensors in everyday objects. In this way, the necessary contextual elements can be captured and interpreted, enabling the construction of sophisticated behaviour models, and facilitating the identification of opportunities where proactive and anticipatory activities by applications and services can be enabled.

4 REALISING AN AMBIENT SOLUTION

Having discussed the broad nature of embedded agents, it is now appropriate to revisit the original motivation for this discussion, namely how to monitor users so as to anticipate their needs and proactively help them in the course of their everyday activities. The solution to this problem does not solely manifest itself either in an exclusively hardware or software solution. Both have an indispensable role to play in fulfilling this objective. WSN technologies represent a significant step forward in the pursuit of the ubiquitous computing vision. However, it must be acknowledged that there are significant technical obstacles outstanding before the transparent integration of computational elements into everyday objects becomes a reality. Power management is one critical issue that springs to mind.

Traditionally, it has been the software element that has been lagging behind hardware developments. However, the increasing incorporation of runtime environments such as Java ME into both mobile devices and embedded technologies is giving rise to a situation where, for the first time, a common platform is available to software developers. Again, it must be acknowledged that there are limitations and incompatibly issues amongst others; nevertheless, the vision of various heterogeneous hardware elements all sharing a common software element is beginning to crystallise. Thus the foundations are laid for realising software services that are inherently distributed yet adaptable to the prevailing context at the time of invocation.

Now that both the hardware and software foundations are in place to begin constructing services that can harness and interpret various aspects of the user's context, the question arises as to how best to engineer such services. Based on the previous discussions, it can be seem that the embedded agent paradigm is a particularly apt one as agents incorporate a significant number of features that can be fruitfully harvested to deliver the necessary adaptivity. This is not to say that it is the only approach or the best approach. It is just an acknowledgment that, at this moment in time at least, embedded agents singularly possess some of the essential characteristics necessary for collecting and interpreting the necessary contextual elements essential to the provision of human-centered services.

To reflect further on some of the more pertinent agent characteristics: autonomy and reactivity are essential to the continuous monitoring of contextual cues. Collaboration is essential for the integration of the contextual cues and the construction of a model of the user's world. Intelligence is necessary for interpreting the meaning of the collated contextual cues and the construction of models of past behaviour that can be used to predict likely future actions. Agents can then proactively use those models to anticipate and pre-empt user requests. Finally, agents are inherently distributed software entities. This makes them ideal for implementing solutions that must harness data from numerous diverse sources, interpreted it in an intelligent and collaborative manner, and collate the results such that an accurate model of the prevailing context at any given time may be constructed. Only in this way can sophisticated behaviour models be constructed, patterns of behaviour identified and future activities predicted, paving the way for the delivery of truly adaptive human centered applications and services.

5 CONCLUSIONS

Embedded agents offer one vision of how disparate data sources may be captured and interpreted to realize services in a range of applications that are human-centric. Their inherent characteristics make them a particularly apt solution for modelling such services. Ongoing developments in WSN technologies are continuously extended the number of platforms on which such agents can be realistically deployed. Yet many challenges remain, including identifying effective strategies for data and decision fusion such that contexts and tasks can be recognised within a time frame that allows effective responses and interventions.

ACKNOWLEDGEMENTS

This material is based upon works supported by the Science Foundation Ireland (SFI) under Grant No. (Grant No. 07/CE/I1147).

REFERENCES

Fok, C., Roman, G., and Lu, C. (2005) Mobile agent middleware for sensor networks: an application case study. In: *Proceedings of the 4th international Symposium on information Processing in Sensor Networks,* Los Angeles, California, April 2005. Piscataway: IEEE Press, pp. 382- 387.

Greenberg, S. (2001) Context as a dynamic construct. *Hum.-Comput. Interact.* 16 (2): 257-268.

Hagras, H., Callaghan, V., Colley, M., Clarke, G., Pounds-Cornish, A., and Duman, H. (2004) Creating an Ambient-Intelligence Environment Using Embedded Agents. *IEEE Intelligent Systems* 19 (6):12-20.

Keegan, S. O'Hare, G.M.P., O'Grady, M.J., (2008) EasiShop: Ambient Intelligence Assists Everyday Shopping. *Information Sciences*, 178 (3):588-611

Kephart, J. O. and Chess, D. M. (2003) 'The Vision of Autonomic Computing'. *Computer* 36 (1): 41-50.

Maybury, M.T. and Wahlster, W. (Eds.). (1998) *Readings in Intelligent User Interfaces*. San Francisco: Morgan Kaufmann Publishers Inc.

Nehmer, J., Becker, M., Karshmer, A., and Lamm, R. (2006) Living assistance systems: an ambient intelligence approach. In: *Proceedings of the 28th international Conference on Software Engineering,* Shanghai, China, May 2006. New York: ACM, pp. 43-50.

O'Hare, G. M.P., O'Grady, M. J., Muldoon, C., and Bradley, J. F. (2006) Embedded agents: a paradigm for mobile services. *Int. J. Web Grid Serv.* 2 (4): 379-405.Vasilakos, A. and Pedrycz, W. (Eds.) (2006) *Ambient Intelligence, Wireless Networking, and Ubiquitous Computing*. Norwood: Artech House, Inc.

O'Grady, M.J., O'Hare, G.M.P. Sas, C., (2005) Mobile Agents for Mobile Tourists: A User Evaluation of Gulliver's Genie. *Interacting with Computers*, 17(4):343-366

O'Hare, G. M. and O'Grady, M. J. (2002) Addressing Mobile HCI Needs through Agents. In *Proceedings of the 4th international Symposium on Mobile Human-Computer interaction,* Pisa, Italy September 2002. London: Springer-Verlag, pp. 311-314.