



| | |
|-------------------------------------|---|
| Title | Intelligent user interfaces for mobile computing |
| Authors(s) | O'Grady, Michael J., O'Hare, G. M. P. (Greg M. P.) |
| Publication date | 2008 |
| Publication information | O'Grady, Michael J., and G. M. P. (Greg M. P.) O'Hare. "Intelligent User Interfaces for Mobile Computing." IGL, 2008. |
| Publisher | IGL |
| Item record/more information | http://hdl.handle.net/10197/1911 |

Downloaded 2024-05-19 21:09:37

The UCD community has made this article openly available. Please share how this access benefits you. Your story matters! (@ucd_oa)



© Some rights reserved. For more information

O'Grady, M.J., O'Hare, G.M.P. Intelligent User Interfaces for Mobile Computing. In: (Ed. Jo Lumsden) Handbook of Research on User Interface Design and Evaluation for Mobile Technology, 318-329 (2008). ISBN: 159904871X. IGI.

Copyright 2008, IGI Global, www.igi-global.com. Posted by permission of the publisher.

Intelligent User Interfaces for Mobile Computing

M. J. O'Grady
Adaptive Information Cluster (AIC)
School of Computer Science & Informatics
University College Dublin,
Belfield,
Dublin 4,
Ireland.

Tel: +353-1-7162922
Fax: +353-1-2697262
Email: Michael.j.ograde@ucd.ie

G.M.P. O'Hare
Adaptive Information Cluster (AIC)
School of Computer Science & Informatics
University College Dublin,
Belfield,
Dublin 4,
Ireland.

Tel: +353-1-7162472
Fax: +353-1-2697262
Email: Gregory.ohare@ucd.ie

Final Draft

Intelligent User Interfaces for Mobile Computing

ABSTRACT

In this chapter, the practical issue of realizing a necessary intelligence quotient for realizing Intelligent User Interfaces (IUIs) on mobile devices is considered. Mobile computing scenarios differ radically from the normal fixed workstation environment that most people are familiar with; and it is in this dynamicity and complexity that the key motivations for realizing IUIs on mobile devices may be found. Thus the chapter initially motivates the need for the deployment of IUIs in mobile contexts by reflecting on the archetypical elements that comprise the average mobile user's situation or context. A number of broad issues pertaining to the deployment of AI techniques on mobile devices are considered before a practical realisation of this objective through the intelligent agent paradigm is presented. It is the authors hope that a mature understanding of the mobile computing usage scenario, augmented with key insights into the practical deployment of AI in mobile scenarios, will aid software engineers and HCI professionals alike in the successful utilisation of intelligent techniques for a new generation of mobile services.

Keywords: Intelligent User Interfaces, mobile computing, artificial intelligence, intelligent agents, ambient Intelligence

INTRODUCTION

Mobile computing is one of the dominant computing usage paradigms at present and encapsulates a number of contrasting visions of how best the paradigm should be realised. Ubiquitous computing (Weiser, 1991) envisages a world populated with artefacts augmented with embedded computational technologies, all linked by transparent high-speed networks, and accessible in a seamless anytime, anywhere basis. Wearable computing (Rhodes, Minar & Weaver, 1999) advocates a world where people carry the necessary computational artefacts about their actual person. Somewhere in between these two extremes lies the average mobile user, equipped with a PDA or mobile phone, and seeking to access both popular and highly specialised services as they go about their daily routine.

Though the growth of mobile computing usage has been phenomenal, and significant markets exist for providers of innovative services, there still exist a formidable number of obstacles that must be surmounted before software development processes for mobile services becomes as mature as current software development practices. It is often forgotten in the rush to exploit the potential of mobile computing that it is radically different from the classic desktop situation; and that this has serious implications for the design and engineering process. The dynamic nature of the mobile user, together with the variety and complexity of the environments in which they operate, provides unprecedented challenges for software engineers as the principles and methodologies that have been refined over years do not necessarily apply, at least in their totality, in mobile computing scenarios.

Quite how to improve the mobile user's experience remains an open question. One approach is concerned with the notion of an application autonomously adapting to the prevailing situation or context in which end-users find themselves. A second approach concerns the incorporation of intelligent techniques into the application. In principle, such techniques could be used for diverse purposes; however, Intelligent User Interfaces (IUIs) represent one practical example where such techniques could be usefully deployed. Thus the objective of this chapter is to consider how the necessary intelligence can be effectively realised such that software designers can realistically consider the deployment of IUIs in mobile applications and services.

BACKGROUND

Research in IUIs has been ongoing for quite some time, and was originally motivated by problems that were arising in standard software application usage. Examples of these problems include information overflow, real-time cognitive overload and difficulties in aiding end-users to interact with complex systems (Höök, 2000); and these problems were perceived as being a by-product of direct-manipulation style interfaces. Thus the concept of the application or user interface adapting to circumstances as they arose was conceived; hence, the terms “adaptive” or “intelligent user interfaces” are frequently encountered in the literature. How to effectively realise interfaces endowed with such attributes is a crucial question and a number of proposals have been put forward. For example, the use of machine learning techniques has been proposed (Langley, 1997) as has the deployment of mobile agents (Mitrovic, Royo, & Mena, 2005).

In general, incorporating adaptivity and intelligence enables applications to make considerable changes for personalisation and customisation preferences as defined by the user and the content being adapted (O'Connor & Wade, 2006). Though significant benefits can accrue from such an approach, there is a subtle issue that needs to be considered. If an application is functioning according to explicit user defined preferences, then it is functioning in a manner that is as the user expects and understands. However, should the system autonomously or intelligently adapt its services based on some pertinent aspect of the observed behaviour of the user, or indeed, based on some other cue, responsibility for the system behaviour moves, albeit partially, from the user to the system. Thus the potential for a confused user or unsatisfactory user experience increases.

A natural question that must now be addressed concerns the identification of criteria that an application might use as a basis for adapting its behaviour. Context-aware computing (Schmidt, Beigl & Gellersen, 1999) provide one intuitive answer to this question. The notion of context first arose in the early 1990s as a result of pioneering experiments in mobile computing systems. Though an agreed definition of context has still not materialised, it concerns the idea that an application should factor in various aspects of the prevailing situation when offering a service. What these aspects might be is highly dependent on the application domain in question. However, commonly held aspects of context include knowledge of the end-user, for example through a user model; knowledge of the surrounding environment, for example through a Geographic Information System (GIS) model; and knowledge of the mobile device, for example through a suitably populated database. Other useful aspects of an end-user's context include an understanding of the nature of the task or activity currently being engaged in, knowledge of their spatial context, that is, location and orientation, and knowledge of the prevailing social situation. Such models can provide a sound basis for intelligently adapting system behaviour. However, capturing the necessary aspects of the end-user's context and interpreting it is frequently a computationally intensive process, and one that may prove intractable in a mobile computing context. Indeed, articulating the various aspects of context and the interrelationships between them may prove impossible, even during system design (Greenberg, 2001). Thus a design decision may need to be made as to whether it is worth working with partial or incomplete models of a user's context. And the benefit of using intelligent techniques to remedy deficiencies in context models needs to be considered in terms of computational resources required, necessary response time and the ultimate benefit to the end-user and service provider.

SOME REFLECTIONS ON CONTEXT

Mobile computing spans many application domains and within these, it is characterised by a heterogeneous landscape of application domains, individual users, mobile devices, environments and tasks (figure 1). Thus developing applications and services that incorporate a contextual component is frequently an inherently complex and potentially time-consuming endeavour; and the benefits that accrue from such an approach should be capable of being measured in some tangible way. Mobile computing applications tend to be quite domain specific and are hence targeted at specific end-users with specialised tasks or objectives in mind. This is in contrast to the one-size-fits-all attitude to general purpose software development that one would encounter in the broad consumer PC arena. For the purposes of this discussion, it is useful to reflect further on the following aspects of the average mobile user's context: end-user profile, devices characteristics, prevailing environment and social situation.

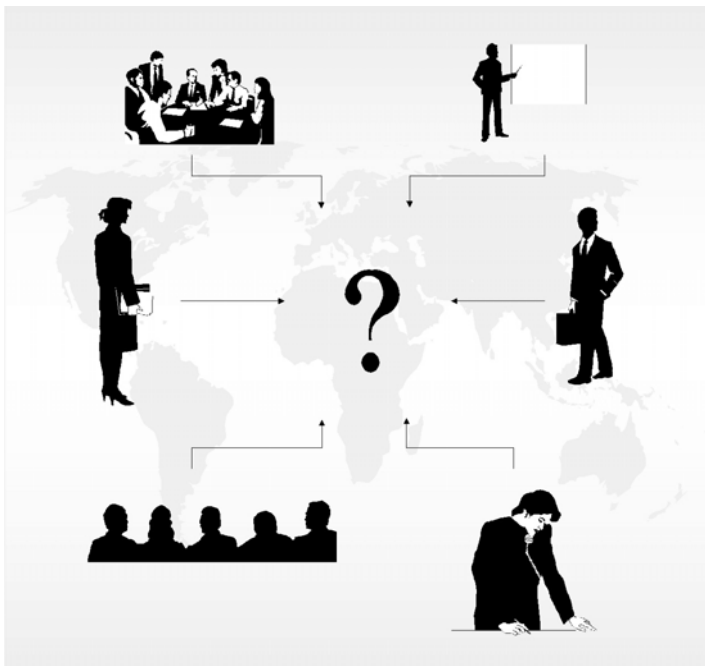


Figure 1: An individual's current activity is a notoriously difficult aspect of an individual's context to ascertain with certainty.

User Profile

Personalisation and customisation techniques assume the availability of sophisticated user models, and currently form an indispensable component of a number of well-known e-commerce related WWW sites. Personalising services for mobile computing users is an attractive proposition in many domains as it offers a promising mechanism for increasing the possibility that the end-users receive content that is of interest to them. Though this objective is likewise shared with owners of e-commerce sites, there are two issues that are of particular importance when considering the mobile user. Firstly, mobile interactions are almost invariably short and to the point. This obliges service providers to strive to filter, prioritise and deliver content that is pertinent to the user's immediate requirements. The second issue concerns the question of costs. Mobile users have to pay for services, which may be charged on a KB basis, thus giving mobile users a strong incentive to curtail their use of the service in question if dissatisfied.

A wide number of features and characteristics can be incorporated into user models. As a basic requirement, some information concerning the user's personal profile, for example, their age, sex, nationality and so on, is required. This basic model may then be augmented with additional sub-models that become increasingly domain-specific. In the case of standard e-commerce services, a record of the previous purchasing history may be maintained and used as a basis for recommending further products. Electronic tourist guides would require the availability of a cultural interest model, which as well as indicating cultural topics of interest to the user, would also provide some metric that facilitated the prioritisation of their cultural interests.

Device Characteristics

Announcements of new devices are occurring with increasing frequency. Each generation successively increases the number of features offered, some of which would not be associated with traditional mobile computing devices, embedded cameras and MP3 players being cases in point. Though offering similar features and services, there are subtle differences between different generations, and indeed interim releases within the same generation, that make the life of a service provider and software professional exceedingly difficult and frequently irritating. From an interface perspective, screen size and support for various interaction modalities are two notable ways in which devices differ, and these have particular implications for the end-user experience. This problem is well documented in the literature and a number of proposals have been put forward to address this, the plasticity concept being a notable example (Thevenin & Coutaz, 1999). Other aspects in which mobile devices differ include processor, memory and operating system; all of which place practical limitations on what is computationally feasible on the device.

Prevailing Environment

The notion of environment is fundamental to mobile computing and it is the dynamic nature of prevailing environment in which the mobile user operates that most distinguishes mobile computing from the classic desktop usage paradigm. As an illustration, the case of the physical environment is now considered, though this in no way diminishes the importance of the prevailing electronic infrastructure. Scenarios in which mobile computing usage can occur are multiple and diverse. Likewise for physical environments. Such environments may be hostile in the sense that they do not lend themselves to easily accessing electronic infrastructure such as telecommunications networks. Other environments may experience extreme climatic conditions thus causing equipment to fail.

Developing a service that takes account of or adapts to the local physical environment is an attractive one. Two prerequisites are unavoidable, however. A model of the environment particular to the service domain in question must be available, and the location of the end-user must be attainable. In the former case, the service provider must construct this environmental model, possibly an expensive endeavour in terms of time and finance. In the latter case, an additional technological solution must be engaged – either one based on satellites, for example GPS, or one that harnesses the topology of the local wireless telecommunications networks. Each solution has its respective advantages and disadvantages, and a practical understanding of each is essential. However, by fulfilling these prerequisites, the service provider is in a position to offer services that take the end-users' physical position into account. Indeed, this vision, often termed location-aware computing (Patterson, Muntz & Pancake, 2003), has grasped the imagination of service providers and end-users alike. In essence, it is a practical example of just one single element of an end-user's context being interpreted and used as a basis for customising services.

Social Situation

Developing a service that adapts to the end-user's prevailing social context is fraught with difficulty, yet is one that many people would find useful. What exactly defines social context is somewhat open to interpretation but in this case, it is considered to refer to the situation in which end-users find themselves relevant to other people. This is an inherently dynamic construct and capturing the prevailing social situation introduces an additional level of complexity not encountered in the contextual elements described previously.

In limited situations, it is possible to infer the prevailing social situation. Assuming that the end-user maintains an electronic calendar, the detection of certain keywords may hint at the prevailing social situation. Examples of such keywords might include lecture, meeting, theatre and so on. Thus an application might reasonably deduce that the end-user would not welcome interruptions, and, for example, proceed to route incoming calls to voicemail and not alert the end-user to the availability of new email. Outside of this, one has to envisage the deployment of a suite of technologies to infer social context. For example, it may be that a device, equipped with a voice recognition system, may be trained to recognise the end-user's voice, and on recognising it, infer that a social situation is prevailing. Even then, there may be a significant margin of error; and given the power limitations of the average mobile device, running a computationally intensive voice recognition system continuously may rapidly deplete battery resources.

ARTIFICIAL INTELLIGENCE IN MOBILE COMPUTING

Artificial Intelligence (AI) has been the subject of much research, and even more speculation, for almost half a century by now. Though failing to radically alter the world in the way that was envisaged, nevertheless, AI techniques have been successfully harnessed in a quite a number of select domains and their incorporation into everyday applications and services continues unobtrusively yet unrelentingly. Not surprising, there is significant interest amongst the academic community in the potential of AI for addressing the myriad of complexity that is encountered in the mobile computing area. From the previous discussion, some sources of this complexity can be easily identified. Resource management, ambiguity resolution, for example, in determining contextual state and resolving user intention in multimodal interfaces, and adaptation, are just some examples. Historically, research in AI has focuses on various issues related to these very topics. Thus, a significant body of research already exists in some of the very areas that can be harnessed to maximum benefit in mobile computing scenarios. A detailed description of these issues may be found elsewhere (Krüger & Malaka, 2004).

One pioneering effort at harnessing the use of intelligent techniques on devices of limited computational capacity is the Ambient Intelligence (AmI) (Vasilakos & Pedrycz, 2006) initiative. AmI builds on the broad mobile computing vision as propounded by the ubiquitous computing vision. It is of particular relevance to this discussion as it is essentially concerned with usability and HCI issues. It was conceived in response to the realisation that as mobile and embedded artefacts proliferate, demands for user attention would likewise increase, resulting in environments becoming inhabitable, or more likely, people just disabling the technologies in question. In the AmI concept, IUIs are envisaged as playing a key role in mediating between the embedded artefacts and surrounding users. However, AmI does not formally ratify the use of any particular AI technique. Choice of technique is at the discretion of the software designer whose selection will be influenced by a number of factors including the broad nature of the domain in question, the requirements of the user, the capability of the available technology and the implications for system performance and usability.

Having motivated the need for AI technologies in mobile contexts, practical issues pertaining to their deployment can now be examined.

STRATEGIES FOR HARNESSING AI TECHNIQUES IN MOBILE APPLICATIONS

It must be reiterated that AI techniques are computationally intensive. Thus, the practical issue of actually incorporating such techniques into mobile applications needs to be considered carefully. In particular, the implications for performance must be determined as this could easily have an adverse effect on usability. There are three broad approaches that can be adopted when incorporating AI into a mobile application and each is now considered.

Network-based Approach

Practically all mobile devices are equipped with wireless modems thus allowing access to data services. In such circumstances, designers can adopt a kind of client/server architecture where the interface logic is hosted on the mobile devices and the core application logic deployed on a fixed server node. The advantage of such an approach is that the designer can adopt the most appropriate AI technologies for the application in question. However, the effect of network latency must be considered. If network latency is significant, the usability of the application will be adversely affected. Likewise, data rates supported by the network in question must be considered. Indeed, this situation is aggravated when it is considered that a number of networks implement a channel sharing system where the effective data rate at a given time is directly proportional to the number of subscribers currently sharing the channel. It is therefore impossible to guarantee an adequate Quality of Service (QoS) making the prediction of system performance difficult. Often, the worst case scenario must be assumed. This has particular implications where the AI application on the fixed server node needs either a significant amount of raw data or a stream of data to process.

One key disadvantage of placing the AI component on a fixed server node concerns the issue of cost. There is a surcharge for each KB of data transferred across the wireless network, and though additional revenue is always welcome, the very fact that the subscriber is paying will affect their perception of application in question and make them more demanding in their expectations.

A network-based AI approach is by far the most common and has been used in quite a number of applications. For example, neural networks have been used for profiling mobile users in conversational interfaces (Toney, Feinberg & Richmond, 2004). InCa (Kadous & Sammut, 2004) is a conversational agent that runs on a PDA but uses a fixed network infrastructure for speech recognition.

Distributed Approach

In this approach, the AI component of the service may be split between the mobile device and the fixed network node. The more computationally expensive elements of the service are hosted on the fixed network node while the less expensive elements may be deployed on the device. Performance is a key limitation of this approach as the computational capacity of the devices in question as well as the data-rates supported by the wireless network can all contribute to unsatisfactory performance. From a software engineering perspective, this approach is quite attractive as Distributed AI (DAI) is a mature research discipline in its own right; and a practical implementation of DAI is the Multi-Agent System (MAS) paradigm.

One example of an application that uses a distributed approach is Gulliver's Genie (O'Grady & O'Hare, 2004). This is a tourist information guide for mobile tourists, realised as a suite of intelligent agents encompassing PDAs, wireless networks and fixed network servers. Agents on the mobile device are

responsible for manipulating the user interface while a suite of agents on the fixed server collaborate to identify and recommend multimedia content that is appropriate to the tourist's context.

Embedded Approach

As devices grow in processing power, the possibility of embedding an AI based application on the actual physical device becomes ever more feasible. The key limitation is performance, which is a direct result of the available hardware. This effectively compromises the type of AI approach that can be usefully adopted. Overtime, it can be assumed that the capability and variety of AI techniques that can be deployed will increase as developments in mobile hardware continue and the demand for ever-more sophisticated applications increases. From an end-user viewpoint, a key advantage of the embedded approach concerns cost as the number of connections required is minimised.

One example of an application that uses the embedded approach is iDorm (Hagras et al, 2004), a prototype AmI environment. This environment actually demonstrates a variety of embedded agents including fixed motes, mobile robots and PDAs. These agents collaborate to learn and predict user behaviour using fuzzy logic principles and, based on these models, the environment is adapted to the inhabitant's needs.

Deployment Considerations

Technically, all three of approaches are viable but the circumstances in which they may be adopted vary. For specialised applications, the networked AI approach is preferable as it offers greater flexibility and maximum performance, albeit at a cost. For general applications, the embedded approach is preferable, primarily due to cost limitations, but the techniques that can be adopted are limited. The distributed approach is essentially a compromise, incorporating the respective advantages and disadvantages of both the networked and embedded approach to various degrees. Ultimately, the nature of the application domain and the target user base will be the major determinants in what approach is adopted. However, in the longer term, it is the embedded approach that has the most potential as it eliminates the negative cumulative effect of network vagrancies, as well as hidden costs. Thus, for the remainder of this chapter, we focus on the embedded approach and consider how this might be achieved.

So what AI techniques can be adopted, given the inherent limitations of mobile devices? Various techniques have been demonstrated in laboratory conditions but one paradigm has been demonstrated to be computationally tractable on mobile devices: intelligent agents. As well as forming the basis of mobile intelligent information's systems, a number of toolkits have been made available under open source licensing conditions thus allowing software engineers access to mature platforms at minimum cost. Before briefly considering some of these options, it is useful to reflect on the intelligent agent paradigm.

THE INTELLIGENT AGENT PARADIGM

Research in intelligent agents has been ongoing since the 1970s. Unfortunately the term agent has been interpreted in a number of ways thereby leading to some confusion over what the term actually means. More precisely, the characteristics that an arbitrary piece of software should possess before applying the term agent to it are debatable. In essence, an agent may be regarded as a computational entity that can act on behalf of an end-user, another agent or some other software artefact. Agents possess a number of attributes that distinguish them from other software entities. These include amongst others:

- Autonomy: the ability to act independently and without direct intervention from another entity, either human or software-related;
- Proactivity: the ability to opportunistically initiate activities that further the objectives of the agent;
- Reactivity: the ability to respond to events perceived in the agent's environment;
- Mobility: the ability to migrate to different nodes of a network as the need to fulfil its objectives dictates;
- Social ability: the ability to communicate with other agents using a shared language and ontology leading to shared or collaborative efforts to achieve individual and shared objectives.

To what extent an agent possesses or utilises each of those attributes is at the discretion of the designer. For clarity purposes, it is useful to consider agents as existing on a scale. At the lower end are so-called reactive agents. Such agents act in a stimulus-response manner, and a typical usage scenario might involve the agent monitoring for user interaction and reacting to it. Such agents are generally classified as weak agents (Wooldridge & Jennings, 1995). At the other end of the scale are so-called strong agents. Such agents maintain a sophisticated model of their environment, a list of goals or objectives, and plans detailing how to achieve these objectives. Such agents support rational reasoning in a collaborative context and are usually realised as Multi-Agent Systems (MAS). This strong notion of agenthood is synonymous with the view maintained by the AI community.

One popular interpretation of the strong notion of agency is that of the Belief-Desire-Intention (BDI) paradigm (Rao & Geogeff, 1995). This is an intuitive and computationally tractable interpretation of the strong agency stance. To summarise: beliefs represent what the agent knows about its environment. Note that the term environment can have diverse meanings here and may not just relate to the physical environment. Desires represent the objectives of the agent, and implicitly the raison d'être for the application. However, at any moment in time, an agent may be only capable of fulfilling some of its desires, if even that. These desires are then formulated as intentions and the agent proceeds to fulfil these intentions. The cycle of updating its model of the environment, identifying desires that can be fulfilled, and realizing these intentions is then repeated for the duration of the agent's lifecycle (figure 2).

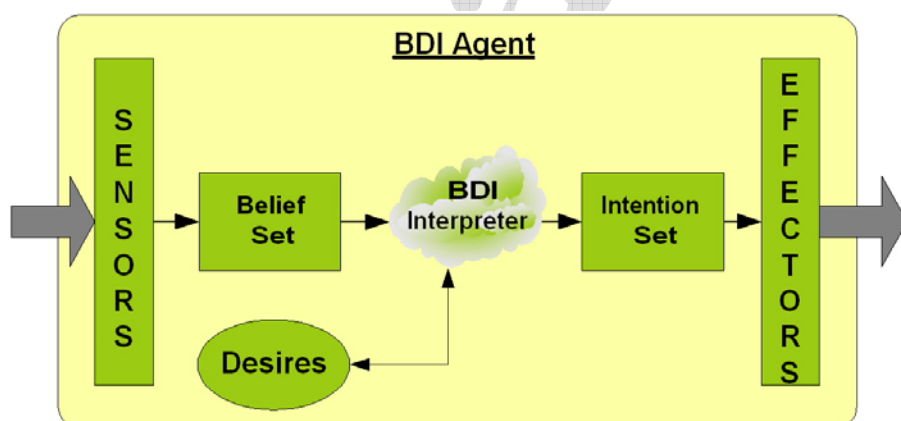


Figure 2: Architecture of a BDI agent.

When should agents be considered for realizing a software solution? Opinion on this is varied. If the solution can be modelled as a series of dynamic interacting components, then agents may well offer a viable solution. However, many see agents as being particularly useful in situations that are inherently complex and dynamic as their native capabilities equip them for handling the myriad of situations that may arise. Naturally, there are many situations that fulfil the above criteria but, for the purposes of this

discussion, it can be easily seen that the mobile computing domain offers significant opportunities for harvesting the characteristics of intelligent agents.

Intelligent Agents for Mobile Computing

As the capability of mobile devices grew, researchers in the intelligent agent community became aware of the feasibility of deploying agents on such devices, and perceived mobile computing as a potentially fertile area for the intelligent agent paradigm. A common approach was to extend the functionality of existing and well-documented MAS environments such that they could operate on mobile devices. It was not necessary to port the entire environment on to the device; it was just necessary to develop an optimised runtime engine for interpreting the agent logic. In this way, the MAS ethos is persevered and such an approach subscribes to the distributed AI approach alluded to previously. A further benefit was that existing Agent-Oriented Software Engineering (AOSE) methodologies could be used. In the case of testing, various toolkits have been released by the telecommunications manufacturers that facilitate the testing of mobile applications. A prudent approach is of course to test the application at various stages during its development on actual physical devices, as this will give a more accurate indication of performance, the Look and Feel (L&F) of the application and so on. For a perspective on deploying agents on mobile devices, the interested reader should consult Carabelea and Boissier (2003).

While a number of environments may be found in the literature for running agents on mobile devices, the following toolkits form a useful basis for initial consideration:

1. LEAP (Lightweight Extensible Agent Platform) (Berger, Rusitschka, Toropov, Watzke & Schichte, 2002) is an extension of the well-documented JADE platform (Bellifemine, Caire, Poggi & Rimassa, 2003). It is FIPA (<http://www.fipa.org/>) compliant and capable of operating on both mobile and fixed devices.
2. MicroFIPA-OS (Laukkanen, Tarkoma & Leinonen, 2001) is a minimised footprint of the FIPA-OS agent toolkit (Tarkoma & Laukkanen, 2002). The original FIPA-OS was designed for PCs and incorporated a number of features that did not scale down to mobile devices. Hence, MicroFIPA-OS minimises object creation, reduces computational overhead and optimizes the use of threads and other resource pools.
3. AFME (Agent Factory Micro Edition) (Muldoon, O'Hare, Collier & O'Grady, 2006) is derived from Agent Factory (Collier, O'Hare, Lowen, & Rooney, 2003), a framework for the fabrication and deployment of agents that broadly conform to the BDI agent model. It has been specifically designed for operation on cellular phones and such categories of devices.
4. JACK is, in contrast to the three previous frameworks, a commercial product from The Agent Oriented Software Group (<http://www.agent-software.com>). It comes with a sophisticated development environment, and like AFME, conforms to the BDI agent model.

A detailed description of the each of these systems is beyond the scope of this discussion. However, the interested reader is referred to (O'Hare, O'Grady, Muldoon & Bradley, 2006) for a more advanced treatment of the toolkits and other associated issues.

FUTURE TRENDS

As mobile devices proliferate, and each generation surpasses its predecessor in terms of raw computational capacity and supported features, the potential for incorporating additional AI techniques will increase. In a similar vein, new niche and specialised markets for mobile services will appear. If a more holistic approach is taken towards mobile computing, it can be seen that developments in sensor technologies, fundamental to the ubiquitous and pervasive vision, will follow a similar trajectory.

Indeed, the possibility of deploying intelligent agents on sensors is being actively investigated in widespread expectation that the next generation of sensors will incorporate processors of a similar capability to the current range of PDAs. Such a development is essential if the AmI vision to reach fruition.

As the possibility of incorporation of ever more sophisticated AI techniques increases, the potential for extending and refining the adaptivity and IUI constructs for the support of mobile users increase. Indeed, adaptivity may reach its fulfilment through the incorporation of autonomic computing precepts (Kephart & Chess, 2003). Self-configuring, self-healing, self-optimising and self-protecting are the key attributes of an autonomic system, and it can be seen that incorporation of AI techniques may make the realisation of these characteristics more attainable.

Finally, the practical issues of engineering mobile AI solutions must be considered. Mobile computing poses significant challenges to the traditional software engineering process, and the broad issue of how best to design for mobile services still needs to be resolved. The situation is exacerbated when AI technologies are included. However, it may be envisaged that as experience and knowledge of the mobile computing domain deepens and matures, new methodologies and best practice principles will emerge.

CONCLUSION

Mobile computing scenarios are diverse and numerous, and give rise to numerous challenges that must be overcome if the end-user experience is to be a satisfactory one. IUIs offers one viable approach that software designers can adopt in their efforts to make their systems more usable in what is frequently a hostile environment. However, the pragmatic issue of realising mobile applications that incorporate intelligent techniques is of critical importance and gives rise to significant technical and design obstacles.

In this chapter, the broad issue of realising an intelligent solution was examined in some detail. At present, the intelligent agent paradigm offers an increasingly viable proposition for those designers who wish to include intelligent techniques in their designs. To illustrate the issues involved, the intelligent agent paradigm was discussed in some detail.

As mobile developments continue unabated, the demand for increasingly sophisticated applications and services will likewise increase. Meeting this demand will pose new challenges for software and HCI professionals. A prudent and selective adoption of intelligent techniques may well offer a practical approach to the effective realisation of a new generation of mobile services.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support of the Science Foundation Ireland (SFI) under Grant No. 03/IN.3/1361.

REFERENCES

Bellifemine, F., Caire, G., Poggi, A., & Rimassa, G. (2003). *JADE - A white paper*. Retrieved January 2007 from <http://jade.tilab.com/papers/2003/WhitePaperJADEEXP.pdf>

O'Grady, M.J., O'Hare, G.M.P. Intelligent User Interfaces for Mobile Computing. In: (Ed. Jo Lumsden) Handbook of Research on User Interface Design and Evaluation for Mobile Technology, 318-329 (2008). ISBN: 159904871X. IGI.

Berger, M., Rusitschka, S., Toropov, D., Watzke, M., & Schichte, M. (2002). Porting distributed agent-middleware to small mobile devices. Proceedings of the Workshop on Ubiquitous Agents on Embedded, Wearable, and Mobile Devices, <http://www.autonomousagents.org/ubiquitousagents/2002/papers>).

Collier, R.W., O'Hare, G.M.P., Lowen, T., & Rooney, C.F.B., (2003). Beyond prototyping in the factory of agents. In J. G. Carbonell and J. Siekmann (Eds.), *Lecture Notes in Computer Science*, 2691, (pp. 383-393), Berlin:Springer.

Carabelea C. & Boissier O. (2003). Multi-agent platforms on smart devices: Dream or reality. Retrieved January 2007 from <http://www.emse.fr/~carabele/papers/carabelea.soc03.pdf>.

Greenberg, S. (2001). Context as a dynamic construct. *Human-Computer Interaction*, 16, 257-268.

Hagras, H., Callaghan, V., Colley, M., Clarke, G., Pounds-Cornish, A., & Duman, H. (2004). Creating an ambient-intelligence environment using embedded agents, *IEEE Intelligent Systems*, 19(6) 12-20.

Höök, K. (2000). Steps to take before intelligent user interfaces become real. *Interacting with Computers*, 12, 409-426.

Kadous, M.W., & Sammut, C. (2004). InCA: A mobile conversational agent, *Lecture Notes in Computer Science*, 3153, (pp. 644-653). Berlin:Springer.

Kephart, J.O., & Chess, D.M. (2003). The vision of autonomic computing. *IEEE Computer*, 36(1), 41-50.

Kruger, A., & Malaka, R. (2004). Artificial intelligence goes mobile, *Applied Artificial Intelligence*, 18, 469-476.

Langley, P. (1997). Machine learning for adaptive user interfaces. *Lecture Notes in Computer Science*, 1303 (pp. 53-62). Berlin:Springer.

Laukkanen, M., Tarkoma, S., & Leinonen, J. (2001). FIPA-OS agent platform for small-footprint devices, *Lecture Notes in Computer Science*, 2333 (pp. 447-460). Berlin:Springer.

Mitrovic, N., Royo, J. A., & Mena, E. (2005). Adaptive user interfaces based on mobile agents: Monitoring the behavior of users in a wireless environment. *Proceedings of the Symposium on Ubiquitous Computation and Ambient Intelligence* (pp. 371-378), Madrid:Thomson-Paraninfo.

Muldoon, C., O'Hare, G.M.P., Collier, R.W., & O'Grady, M.J. (2006). Agent factory micro edition: A framework for ambient applications, *Lecture Notes in Computer Science*, 3993(pp. 727-734). Berlin:Springer.

O'Connor, A. and Wade, V., (2006). Informing context to support adaptive services, *Lecture Notes in Computer Science*, 4018 (pp.366-369). Berlin:Springer.

O'Grady, M.J., & O'Hare, G.M.P. (2004). Just-in-Time multimedia distribution in a mobile computing environment, *IEEE Multimedia*, 11(4), 62-74.

O'Grady, M.J., O'Hare, G.M.P. Intelligent User Interfaces for Mobile Computing. In: (Ed. Jo Lumsden) Handbook of Research on User Interface Design and Evaluation for Mobile Technology, 318-329 (2008). ISBN: 159904871X. IGI.

O'Hare, G.M.P., O'Grady, M.J., Muldoon, C., & Bradley, J.F. (2006). Embedded agents: A paradigm for mobile services, *International Journal of Web and Grid Services*, 2 (4), 355-378.

Patterson, C.A., Muntz, R.R. & Pancake, C.M. (2003). Challenges in location-aware computing. *IEEE Pervasive Computing*, 2(2), 80-89.

Rao, A.S., & Georgeff, M.P. (1995). BDI agents: from theory to practice. In V. Lesser and L. Gasser (Eds.), *Proceedings of the First International Conference on Multiagent Systems* (pp. 312-319). California:MIT Press.

Rhodes, B.J., Minar, N. & Weaver, J. (1999). Wearable computing meets ubiquitous computing: Reaping the Best of Both World. *Proceedings of the Third International Symposium on Wearable Computers* (pp. 141-149). California:IEEE Computer Society

Schmidt, A., Beigl, M. & Gellersen, H-W, (1999). There is more to context than location. *Computers and Graphics*, 23(6), 893-901.

Tarkoma, S., & Laukkanen, M. (2002). Supporting software agents on small devices. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)* (pp. 565-566). New York:ACM Press.

Thevenin, D., & Coutaz, J. (1999). Plasticity of user interfaces: Framework and research agenda. In (M. A. Sasse & C. Johnson (Eds.), *Proceedings of IFIP TC 13 International Conference on Human-Computer Interaction (INTERACT'99)*, (pp. 110-117). Amsterdam:IOS Press.

Toney, D., Feinberg D., & Richmond, K. (2004). Acoustic features for profiling mobile users of conversational interfaces, *Lecture Notes in Computer Science*, 3160 (pp. 394-398). Berlin: Springer.

Vasilakos, A. and Pedrycz, W. (2006). *Ambient Intelligence, Wireless Networking, Ubiquitous Computing*, Norwood:Artec House.

Weiser, M. (1991). The computer for the twenty-first century, *Scientific American*, 265(3), 94-100.

Wooldridge, M., & Jennings, N.R. (1995). Intelligent agents: Theory and practice, *Knowledge Engineering Review*, 10(2), 115-152.

↑

TERMS AND DEFINITIONS

Ambient Intelligence

Ambient Intelligence (AmI) was conceived by the Information Society Technologies Advisory Group (ISTAG) as a means of facilitating intuitive interaction between people and ubiquitous computing environments. A key enabler of the AmI concept is the Intelligent User Interface.

BDI architecture

The Belief-Desire-Intention (BDI) architecture is an example of a sophisticated reasoning model based on mental constructs that can be used by intelligent agents. It allows the modelling of agents' behaviours in an intuitive manner that complements the human intellect.

Context

Context-aware computing considers various pertinent aspects of the end-user's situation when delivering a service. These aspects, or contextual elements, are determined during invocation of the service and may include user profile, for example language, age, and so on. Spatial contextual elements, namely location and orientation, may also be considered.

Intelligent Agent

Agents are software entities that encapsulate a number of attributes including autonomy, mobility, sociability, reactivity and proactivity amongst others. Agents may be reactive, deliberative or hybrid. Implicit in the agent construct is the requirement for a sophisticated reasoning ability, a classic example being agents modelled on the BDI architecture.

Intelligent User Interface

Intelligent User Interfaces harness various techniques from Artificial Intelligence to adapt and configure the interface to an application such that the end-user's experience is more satisfactory.

Mobile Computing

Mobile computing is, in essence, a computer usage paradigm where end-users access applications and services in diverse scenarios, while mobile. Mobile telephony is a popular realisation of this paradigm, but wearable computing and telematic applications could also be considered as realistic interpretations of mobile computing.

Multi-Agent System

A suite of intelligent agents, seeking to solve some problem beyond their individual capabilities, come together to form a Multi-Agent System (MAS). These agents collaborate to fulfil individual and shared objectives.

Ubiquitous Computing

Conceived in the early 1990s, Ubiquitous Computing envisages a world of embedded devices, where computing artefacts are embedded in the physical environment and accessed in a transparent manner.