2010 Symposia and Workshops on Ubiquitous, Autonomic and Trusted Computing

Personal Smart Spaces as a Basis for Identifying Users in Pervasive Systems

Elizabeth Papadopoulou, Sarah Gallacher, Nick K. Taylor and M. Howard Williams
School of Maths and Computer Sciences,
Heriot-Watt University,
Edinburgh EH14 4AS, UK
{E.Papadopoulou, S.Gallacher, N.K.Taylor, M.H.Williams}@hw.ac.uk

***Abstract*— The notion of a Personal Smart Space (PSS) offers a new and flexible solution to the problem of implementing pervasive systems. Its attractiveness lies not only in the way in which it bridges the gap between conventional fixed smart spaces and mobile ubiquitous systems, but also in the new functionality it can provide to enhance the user experience. The use of PSSs to realize pervasive systems is currently being investigated in the Persist project, which has developed a pervasive system platform through which these ideas are being explored. This paper considers one of the features of PSSs, namely that of identifying users through the interaction of their respective PSSs, and the problems that arise when dealing with multiple Personal Smart Spaces. Some scenarios are presented to illustrate how this might be utilized in novel services. The problems surrounding implementation are discussed and a solution based on the Persist architecture is presented.**

Keywords- Pervasive systems, smart spaces, ubiquitous systems, user identification, personalisation.

I. INTRODUCTION

As the numbers of devices and networks continue to grow rapidly, the vision of an environment surrounding the user that is filled with microscopic devices, mobile or stationary, that will aid the user in his/her everyday life (Weiser [1]) is becoming more and more of a reality. This growth is accompanied by an even larger expansion in the services available to a user, and the result will soon be unmanageable. Consequently the need to support the user in controlling the resulting growing complexity is becoming increasingly urgent. At the same time the amount of research being done on pervasive [2] and ubiquitous computing is growing to meet this challenge [3, 4], and more and more prototypes are emerging to test different subsets of ideas in this area.

The development of prototype pervasive systems has also led to the emergence of new functionality that can be used to enhance the user experience. One challenge is now to explore these new functionalities and the pervasive services that can be developed with them to strengthen the user experience and the range of support that can be handled.

The notion of a Personal Smart Space (PSS) is a useful and potentially powerful concept in the development of pervasive system architectures. This concept is based on an ad hoc network which may interact with other PSSs when these are encountered. A particular advantage of this approach is that, because of their peer to peer nature, such smart spaces can be deployed without relying on fixed infrastructure provided by Internet Service Providers or Mobile Network Operators. Thus users can deploy their own personal smart spaces, populating them with their mobile and fixed devices.

Using a PSS as a basis for pervasive systems provides a natural way to bridge the gap between conventional fixed smart spaces and mobile ubiquitous systems. It also provides new functionality that can be used to enhance the user experience.

The Persist project is a European research project funded under Framework 7, which is investigating the use of Personal Smart Spaces (PSSs) as a basis for a pervasive system architecture. It has developed a pervasive system platform based on PSSs to experiment with the ideas, and is currently evaluating this.

In addition to studying its suitability as a basis for implementing pervasive systems, Persist is also studying the functionality of Personal Smart Spaces. PSSs may be mobile or fixed and may interact with each other as the mobile ones move around. This has some unique features that can provide useful functionality that is not easily available in other pervasive systems. Thus Persist is seeking to identify advantages that this approach can bring to pervasive systems in terms of the functionality it might support and the services that might emerge from using this functionality.

This paper is concerned with one such area – namely, the identification of other PSSs. This functionality is an important feature of the PSS architecture and provides a basis for a variety of services that can support a range of different types of users. In particular, the paper focuses on its potential for identifying other users through their PSSs and using this to provide support for the user. This forms an essential part of the prototype that has been developed and will be demonstrated in September 2010.

To use this, it is essential that the system should support two basic functions. Firstly, it must be capable of pro-active behaviour, i.e. it must be able to take actions for the user without waiting for the user to request them. Secondly, it must know what the user would prefer, i.e. it must maintain a set of user preferences for each user and use these to personalize the decision making processes.

This work is supported by the European Union under the FP7 programme (Persist project).

The next section provides a brief background to pervasive systems. Section 3 gives a brief overview of the notion of Personal Smart Spaces while section 4 presents some scenarios to illustrate the problem to be addressed. Section 5 describes the structure of a PSS in Persist and Section 6 covers some of the problems encountered with implementing this functionality. Section 7 describes a potential solution based on the PSS approach. Section 8 concludes.

II. BACKGROUND

Much research has been devoted to the development of ubiquitous or pervasive systems over the past decade. In seeking to develop such systems, different projects have adopted different assumptions and investigated different approaches. From this two main classes of system are of particular interest.

One class of system that can be identified is that concerned with fixed smart spaces. This type of system is concerned with the development of intelligent buildings, of which the most important is the Smart Home. Here the focus has been on developing techniques to support intelligent building automation, which will provide automatic control of devices providing lighting, temperature control, security, etc. The main motivation for much of this has been to provide support for elderly and disabled residents, making it safe for them to live at home. Examples include the Adaptive House [5], MavHome [6], GAIA [7], Synapse [8], Ubisec [9], etc.

In general the notion of a smart space can be defined as "*a multi-user, multi-device, dynamic interaction environment that enhances a physical space by virtual services*" [10, 11]. The services are the means of interaction between participants, objects and the smart spaces. Essentially such smart spaces are based on infrastructure and sensor-equipped rooms.

On the other hand the other major type of system that has attracted attention is one that aims to address the needs of the mobile user. Mobility presents different and more challenging problems than those of fixed smart spaces. Here the aim is to provide access to devices, networks and services wherever the user may be. By its very nature such systems need to be context-aware and select services that are appropriate to a user's current context and needs. Thus, for example, different services might be selected if the user is at work compared with those selected at home, in town or travelling in a car. Likewise different network and device options may be selected depending on resources available in the user's current environment and context.

A number of research projects have explored pervasive system architectures for the mobile user and a number of prototypes have been developed to demonstrate these, for example Mobilife [12], Spice [13], etc. Another European research project to develop architectures for pervasive systems was Daidalos [14]. This project explored two separate architectures [15] for pervasive systems, focussing particularly on mobile users, and developed prototypes for each of these.

In order to be acceptable to the end user, it is essential that pervasive systems are adaptive to the needs of the individual user and personalise their behaviour according to the needs and preferences of different users and the particular context which the user is in. For this purpose some form of knowledge must be held about the user's preferences and behaviour patterns and must be applied when the appropriate decisions are taken. This may be done proactively by identifying what actions the user might wish to take and performing these actions on the user's behalf.

The simplest approach to handling such knowledge is through the use of user preferences based on rules. Initial systems using this approach made the assumption that such preferences would be entered manually by the user. However, building up a realistic set of preferences in this way is very time consuming and experience has shown that the user soon loses interest and the resulting preference sets are incomplete and not very useful. As a result systems sought alternative approaches such as monitoring of the user's behaviour followed by some form of learning (e.g. the fixed smart space MavHome [6] project, or mobile applications, e.g. Specter [16]). Another alternative is to use other forms of knowledge representation such as Bayesian networks or Hidden Markov Models rather than rule based preferences to capture user behaviour and represent user needs (e.g. Synapse project [8]).

III. PERSONAL SMART SPACES

The notion of a Personal Smart Space (PSS) [17] has been proposed as a way of bridging the gap between conventional fixed smart spaces and pervasive systems developed for the mobile user. While the vision of many fixed smart space projects is to provide increasing levels of support for the user who inhabits the fixed space, when he/she steps outside of this island, all of this is lost. The result is that one will end up with islands of pervasiveness separated by voids in which support for pervasiveness is limited. The idea behind the PSS is that the user will be constantly covered by their own pervasive PSS, although the facilities it can offer at any point in time will vary depending on other PSSs in the neighbourhood.

A PSS is defined by the set of services that are running or available within a dynamic space of connectable devices where the set of services and devices are owned, controlled, or administered by a single user or organisation. It can be realised as an ad hoc network which may interact with the networks of other PSSs when these are encountered. This has the advantage of not requiring any fixed infrastructure to be provided by Internet Service Providers or Mobile Network Operators, although it is able to take advantage of infrastructure when it is available. Thus users can deploy their own personal smart spaces, populating them with their mobile and fixed devices.

A PSS has a number of important characteristics, including the following:

- The set of devices and services that make up the PSS have a single owner, whether this is a person or an organisation.
- A PSS may be mobile or fixed.
- It must be able to adapt to different situations depending on the context and user preferences.
- It must be self-improving and able to learn from monitoring the user to identify trends, and infer

conditions when user behaviour or preferences change.

Persist [18] is a European collaborative research project which is investigating the idea of personal smart spaces as a basis for the architecture of a pervasive system.

IV. Some Scenarios

One very useful functionality in a pervasive system is the ability to identify other people in the near environment. Weiser [1] includes the idea indirectly in his original paper when Sal looks out of her window and "sees electronic trails that have been kept for her of neighbours coming and going during the early morning". Although he does not elaborate on the technology needed to achieve this, the implication from what follows is that video streams are analysed and image processing used to identify the individuals who passed by.

The idea of using image processing to identify people in the vicinity of the user has also been proposed by others although the practicality of this is challenging. An alternative approach used in some science fiction works (such as the film Minority Report) is to use iris scanning – with consequential illicit trade in eye transplants!

However, the PSS approach opens up an alternative and more practical way of achieving this functionality and raises the possibility of a range of new types of services based on the interactions between PSSs. This may be one mobile PSS encountering another, or a mobile PSS encountering a fixed one.

Communication with the user may occur in various ways – for example, if the user is wearing an earpiece for a mobile phone, the system could communicate with the user using a text-to-voice translator. Alternatively, there has been interest in producing spectacles with a display built into them, which projects the screen onto the lenses so that the user can see the screen while looking through it to the world beyond. This type of device is already commercially available although the technology is still very limited.

For example, consider a situation in which a user X encounters another user Y. Using the PSS as a memory aid, X's PSS could provide a reminder to X of Y's name and other details. This may be done via an earpiece or through these special spectacles and in either case the other party, Y, would be unaware of the communication that was taking place.

To illustrate the potential of this, consider a few examples from the life of Ann, a lecturer in Computer Science. At the end of her first lecture to the large first-year class, one of the students approaches her with a question. Immediately her PSS identifies the student and tells Ann the student's name and some key details on the student (previous record, disabilities, etc.).

On another occasion she passes a student in the corridor who has failed to hand in an assignment for her. Again her PSS registers this and alerts her to the situation and displays the name of the student along with appropriate relevant details (e.g. the class concerned).

Later she goes to a conference to present a paper on her research and to keep her up to date in the latest developments in the area. When she enters the conference hall, there are hundreds of other delegates already there. Through her PSS she is able to locate other delegates whom she already knows. She is also able to locate other delegates from the same country, or who are working in the same area as she is. She may do this through direct contact with their PSSs provided they have made their information accessible to her, or indirectly through the conference PSS.

One can envisage a range of similar scenarios in other business situations where the user meets with clients, either planned or unexpected, or works with colleagues in different parts of a company or from different companies.

There are also particular advantages for disabled users. For example, it could assist someone in the early stages of dementia who is coping with his condition by and large but does need help remembering people. When he meets Bill, his PSS identifies Bill and tells him Bill's name through his earpiece while displaying some pictures of Bill from the past to help trigger his memory.

Although each of these scenarios relies on slightly different functionality provided by third party applications – a student identification service for academic staff (coupled to the student record system), a general business client identification service, a conference service and a personal elderly support service – they are all based on the same principle of identification of PSSs through their interactions.

The same functionality can be used between a mobile PSS and a fixed one to support a range of other applications. An obvious example is that of controlling access to buildings, offices, laboratories, etc. This could extend to meeting rooms to which access is permitted to those involved for the duration of the meeting. And so on. This could even extend to greeting a user as he/she walks through the door (again illustrated in Minority Report). However, the functionality required in each case is identical to that of the mobile-to-mobile case.

V. Structure of a Personal Smart Space in Persist

This section provides a brief introduction to the PSS high level architecture design adopted in the Persist system. At this level the architecture can be viewed as consisting of five layers, each of which incorporates various component blocks and components that are essential to the design of the PSS environment. This architecture is illustrated in Figure 1. Each layer addresses a well defined part of the PSS functionality. These are described in more detail below.

Layer 1 - Devices

At the lowest level of the architecture one has the set of devices making up the PSS. These devices vary in their processing and networking capabilities. In particular, they may include:

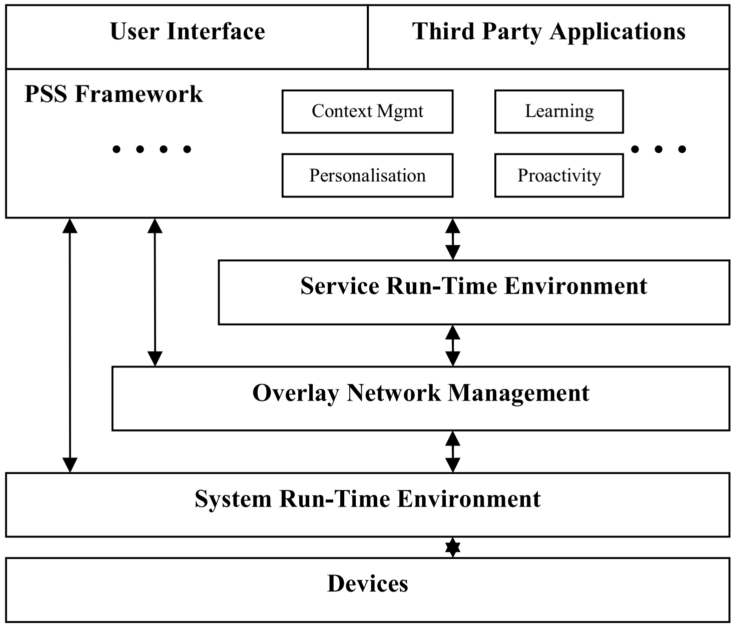

Fig. 1. The high level architecture of a Personal Smart Space

(i) *servers* (i.e. independent computers dedicated to provide one or more services over a computer network) such as Windows Media Center, Apple Itheatre, PCs, etc.,

(ii) *laptops* (i.e. small-sized portable computers) such as Mac Book, Sony Vaio, Tablet PC, etc.,

(iii) *mobile phones* (i.e. pocket-sized handheld computing devices) such as iPhone, HTC Tytan, Nokia N90, PDAs, etc.,

(iv) *sensors* (i.e. group of devices that may be embedded into other devices, can measure a physical parameter and convert it into a signal, which in turn can be read by an observer or an instrument); examples include RFID readers, GPS location estimators, accelerometers, thermometers, altimeters, barometers, etc.

(v) *smart objects* (i.e. resource-constrained devices that can be connected to the Internet or a LAN via a wifi connection, ethernet, GPRS, 3G, etc., usually intended for displaying multimedia content such as a combination of text, audio, still images, animation and video or other everyday objects enhanced with pervasive facilities); Examples include WiFi photoframes, Chumby, Nabaztag, home eAppliances, surveillance cameras, etc., and

(vi) *interactive entertainment electronic devices* (i.e. an electronic device producing a video display signal, which can be used with a display device such as a television set or monitor to display a video game or an external source of signal). A gaming console is an example.

Depending on their processing and networking capabilities, each of these devices may implement a subset of the PSS stack (PSSlite) or all of it, or simply interact with the rest of the PSS framework.

Layer 2 - System Run-Time Environment

The purpose of the System Run-Time layer is to provide platform independence as far as possible by acting as an abstraction layer between the operating system of the device and the higher level PSS software. By this means this layer is responsible for making a device PSS-enabled. Thus by using an “off-the-shelf” implementation of a virtual machine run-time environment one can achieve PSS portability over a wide range of software and hardware platforms. Also included in this layer is responsibility for device mobility and sensor management.

Layer 3 - Overlay Network Management

The Overlay Network Management layer is responsible for Peer-to-Peer (P2P) management and communication, both within the PSS itself and between PSSs. More specifically, services within this layer include the functionality for PSS peer group management, PSS peer discovery, peer PSS group discovery, PSS communication management and message routing between peer networks of PSSs. The scope of Persist excludes the lower level ad-hoc networking functionality, and it has been assumed that this will be managed by third party components outside the scope of the Persist framework.

Layer 4 - Service Run-Time Environment

The Service Run-Time Environment layer provides a container for the PSS services. It supports service life cycle management features and provides a service registry, as well as, a device registry. Moreover, it allows for service management in a distributed fashion across multiple devices within the same PSS. In this context, it delivers fault tolerance as well as device resource management. The Service Run-Time Environment also provides advanced information management features for achieving high availability of data, for addressing storage requirements of PSS services, and for supporting event and message management.

Layer 5 - PSS Framework

The PSS Framework layer is the core of the PSS architecture. Its functionality includes service discovery, composition and session management (both PSS and 3rd party services) as well as management of context information, including user preferences. Moreover, the PSS Framework layer supports inference of context information, automatic learning of preferences, and identification of user's future intentions. This information, together with data provided by the recommender system of this layer, enables the proactive facilities of the PSS platform. The PSS Framework layer also offers support for user interaction monitoring as well as user feedback collection and management. Furthermore, this layer provides support for conflict resolution, grouping of context data and preferences and resource sharing. Finally, the PSS Framework layer enables security and privacy management, demonstrating features such as access control, identity management, privacy and trust management, and policy management. However, it should be mentioned that some security and privacy facilities of the PSS also need support from layers 2, 3 and 4 to enable a fully secure and privacy-aware PSS system.

VI. Problems Associated with PSS Identification

When a new PSS is detected in the user's neighbourhood, one is faced with three problems:

(a) Is the PSS of interest?

(b) Where is it located?

(c) What should one do with it?

The first problem - determining whether or not a PSS is of interest - is a complex and challenging one. This depends on the user's current context and the user's preferences.

To illustrate the problem consider as an example the case of a lecturer (Ann) encountering a student. In the university she may be frequently surrounded by students on occasions during the day. In this case relative location within the University is important. If a student enters her office to speak to her, this is clearly relevant and easy to identify. If the student approaches her in the lecture room at the end of a lecture to ask a question, once again both location and current activity are important. If the student passes her in the corridor, she will not want to be alerted unless there is a particular problem relating to the student – e.g. work not handed in for one of her courses.

From this it should be clear that identifying another PSS is only part of the problem. Current context needs to be taken into account to determine whether or not a particular PSS is relevant.

The second question is closely linked to the first, and concerns how close the other PSS is and where it is in relation to the user. Thus in the case of the lecturer in the lecture room, it is only the student who is immediately in front of the lecturer who is of interest. It is a far more challenging problem (though not impossible) to identify other students in the audience.

On the basis of the answers to the previous two questions some action may need to be taken. In both of the above cases (work colleague or student) his/her name and details may be required. The amount of detail and the way in which these are presented (earpiece, spectacles, etc.) depends on user preference.

VII. The Persist Approach

One of the main assumptions made in developing the Persist platform is that the privacy of the individual needs to be protected as far as possible. To achieve this we have adopted an approach based on Digital Personal Identifiers (DPIs). A DPI is a simple virtual identifier that is used as a user name to conceal the actual identity of the user where necessary. Each PSS owner must be assigned a unique Digital Personal Identifier (DPI), which is the most general form of identifier available. The process through which DPIs are allocated is beyond the scope of this paper.

The way in which PSSs may communicate with one another is via their ad hoc networks. To begin with, when the ad hoc network of one PSS comes within range of the network of another PSS, each PSS must identify itself using its DPI. In the process the network can be reconfigured to include both PSSs. A similar situation arises if a third PSS approaches the linked pair resulting in a further reconfiguration of the network to embrace all three.

While a PSS is connected to one or more other PSSs, it may do any of the following:

(1) It may make available any of its services for other PSSs to share. For example, in the case of a fixed PSS acting as a smart home or smart office, the PSS may make available appropriate services to the mobile PSSs of the users that are present in it at any point in time. Even mobile users may wish to make available some of their services for other PSSs (mobile users or fixed PSSs) to access.

(2) It may use any of the services offered by other PSSs in the network.

(3) It may recognize the DPI of another PSS and use this to alert its owner.

(4) It may obtain more information about the other PSS from it directly if it is prepared to divulge this.

In order to control access to information and services, it is useful to distinguish three cases:

(1) Universal access. In this case access is open to any other PSS. An example of this would be the information services associated with some fixed PSSs.

(2) Individual access. Here access is restricted to a particular PSS. This would be the case where one user allows one other specified user to access one of his/her services.

(3) Group access. This is the case where one user is prepared to grant access to any users belonging to a named group. An example might be if the user creates a group of his/her friends by specifying their DPIs to the system.

The particular case that this paper is concerned with is the identification of another PSS and alerting the user to details about the other PSS. Because of the different types of response that might be required for different types of user, this is handled by a third party service.

The Persist platform itself is responsible for dealing with the interaction with another PSS but when such a PSS is identified, a third party service is invoked to take the necessary action and provide the user with the information on the newly encountered PSS.

From the scenarios outlined earlier different third party services may be required for different types of applications. The appropriate service is responsible for determining what information to provide to the user and how to present it – e.g. converting from text to speech and communicating with the user via an earpiece.

VIII. SUMMARY AND CONCLUSION

Personal Smart Spaces have a number of features that can enhance the architecture of a pervasive system by providing novel functionality. This paper considers one of these, namely the identification of other users through interaction with their PSSs. This could lead to a number of new services to support different types of users.

This paper provides an outline of the idea and describes some scenarios that could make use of this functionality. It discusses some of the problems that need to be resolved; in particular, one of the major challenges is that of recognising that a PSS is in the vicinity and identifying it.

The paper also gives a brief outline of how PSS identification is handled within the Persist framework. The approach described here has been implemented in the Persist pervasive system, and will be demonstrated in September 2010.

ACKNOWLEDGMENT

Acknowledgments. The authors wish to thank all colleagues in the Persist project developing Personal Self-Improving Smart Spaces. However, it should be noted that this paper expresses the authors' personal views, which are not necessarily those of the Persist consortium. Apart from funding the Persist project, the European Commission has no responsibility for the content of this paper.

REFERENCES

[1] M. Weiser, "The computer for the 21st century," Scientific American vol. 265(3), pp. 94-104, 1991.

[2] M. Satyanarayanan, "Pervasive computing: vision and challenges," IEEE PCM, vol. 8(4), pp. 10-17, 2001.

[3] J. Sun, "Mobile ad hoc networking: an essential technology for pervasive computing," *Proc. Int Conf on Info-tech & Info-net*, Beijing, China, 2001, pp. 316-321.

[4] A. Zaslavsky, "Adaptability and interfaces: key to efficient pervasive computing," *NSF Workshop on Context-Aware Mobile Database Management*, Providence, Rhode Island, 2002, pp. 24-25.

[5] M. C. Mozer, "Lessons from an Adaptive House," Smart Environments: Technologies, protocols and applications, D. Cook & R. Das, Eds., 2004, pp. 273-294.

[6] M. G. Youngblood, L. B. Holder, and D. J. Cook, "Managing Adaptive Versatile Environments," *Proc. 3rd IEEE Int. Conf. on Pervasive Computing and Communications (PerCom '05)*, 2005, pp. 351-360.

[7] B. D. Ziebart, D. Roth, R. H. Campbell, and A. K. Dey, "Learning Automation Policies for Pervasive Computing Environments," *Proc. 2nd Int. Conf. on Autonomic Computing (ICAC '05)*, 2005, pp. 193-203.

[8] V. Lesser, M. Atighetchi, B. Benyo, B. Horling, A. Raja, R. Vincent, T. Wagner, P. Xuan, and S. X. Q. Zhang, "The Intelligent Home Testbed," Anatomy Control Software Workshop, 1999, pp. 291-298.

[9] K. Gopalratnam and D. J. Cook, "Online Sequential Prediction via Incremental Parsing: The Active LeZi Algorithm," IEEE Intelligent Systems, vol. 22(1), pp. 52-58, 2007.

[10] M. Román, C. K. Hess, R. Cerqueira, A. Ranganathan, R. H. Campbell, and K. Nahrstedt, "Gaia: A middleware infrastructure to enable active spaces," IEEE Pervasive Computing, vol. 1, pp. 74–83, 2002.

[11] B. Johanson, A. Fox, and T. Winograd, "The interactive workspaces project: Experiences with ubiquitous computing rooms," IEEE Pervasive Computing, vol. 1, pp. 67–74, 2002.

[12] T. Kindberg and J. Barton, "A web-based nomadic computing system," Computer Networks, vol. 35, pp. 443–456, 2001.

[13] H. Si, Y. Kawahara, H. Morikawa, and T. Aoyama, "A stochastic approach for creating context aware services based on context histories in smart Home," Proc. 1st International Workshop on Exploiting Context Histories in Smart Environments, 3rd Int Conf on Pervasive Computing (Pervasive 2005) , 2005, pp. 37-41.

[14] J. Groppe and W. Mueller, "Profile Management Technology for Smart Customizations in Private Home Applications," *Proc 16th Int. Workshop on Database and Expert Systems Applications (DEXA'05)*, 2005, pp. 226-230.

[15] T. Kindberg, et al., "People, places, things: Web presence for real world," *Proc. 3rd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA 2000)*, Los Alamitos, Calif., IEEE CS press, 2000, pp. 19 – 28.

[16] G. D. Abowd and E. D. Mynatt, "Designing for the human experience in smart environments," Smart Environments, D.J. Cook, and S.K. Das, editors, Wiley, 2005, pp. 153-174.

[17] M. Strutterer, O. Coutand, O. Droegehorn, and K. David, "Managing and Delivering Context-Dependent User Preferences in Ubiquitous Computing Environments," *Proc. Int. Symp. on Applications and the Internet Workshops (SAINTW '07)*, 2007.

[18] C. Cordier, F. Carrez, H. Van Kranenburg, C. Licciardi, J. Van der Meer, A. Spedalieri, J. P. Le Rouzic, and J. Zoric, "Addressing the Challenges of Beyond 3G Service Delivery: the SPICE Service Platform," *Proc. Workshop on Applications and Services in Wireless Networks (ASWN '06)*, 2006.

[19] M. H. Williams, N. K. Taylor, I. Roussaki, P. Robertson, B. Farshchian, and K. Doolin, "Developing a Pervasive System for a Mobile Environment," Proc. eChallenges 2006 – Exploiting the Knowledge Economy, IOS Press, 2006, pp. 1695 – 1702.

[20] S. McBurney, E. Papadopoulou, N. K. Taylor, M. H. Williams, and Y. Abu Shabaan, "Comparing Two Different Architectures for Pervasive Systems from the Viewpoint of Personalisation," Proc. eChallenges 2009, IOS Press, 2009, (ISBN: 978-1-905824-13-7), 8pp.

[21] A. Kroner, D. Heckmann, and W. Wahlster, "SPECTER: Building, Exploiting, and Sharing Augmented Memories," Proc. Workshop on Knowledge Sharing for Everyday Life (KSEL06), 2006.

[22] M. Crotty, N. Taylor, M. H. Williams, K. Frank, I. Roussaki, and M. Roddy, "A Pervasive Environment Based on Personal Self-Improving Smart Spaces," Proc. Workshop on Constructing Ambient Intelligence, Ambient Intelligence 2008, Springer Verlag, 2010, pp. 58-62.

[23] Persist project homepage: http://www.ict-persist.eu, accessed on 12th July 2010.