Managing Diversity in Practical Ambient Assisted Living Ecosystems

M.J. O'Grady[1,*], M. Walsh [2] , R. Tynan [1], C. Gurrin[3], M. Dragone [1] , and G.M.P. O'Hare [1]

[1] University College Dublin
[2] Tyndall National Institute
[3] Dublin City University

E-Mails: michael.j.ogrady@ucd.ie; michael.walsh@tyndall.ie; cgurrin@computing.dcu.ie; Richard.Tynan@ucd.ie; Mauro.Dragone@ucd.ie; gregory.ohare@ucd.ie.

[*] Author to whom correspondence should be addressed; Tel.: +353-1-7162-922; Fax: +353-1-269-7262

Abstract: Though the motivation for developing Ambient Assisted Living (AAL) systems is incontestable, significant challenges exist in realizing the ambience that is essential to the success of such systems. By definition, an AAL system must be omnipresent, tracking occupant activities in the home and identifying those situations where assistance is needed or would be welcomed. Embedded sensors offer an attractive mechanism for realizing ambience as their form factor and harnessing of wireless technologies aid in their seamless integration into pre-existing environments. However, the heterogeneity of the end-user population, their disparate needs and the differing environments in which they inhabit, all pose particular problems regarding sensor integration and management

Keywords: Ambient Assisted Living; Pervasive Heath; Middleware.

1. Introduction

For an arbitrary AAL environment, a judicious selection of sensors is required to meet individual requirements. These may include, for example, contact and pressure sensors, or motion sensors, possibly of the Passive Infrared (PIR) type. Even a relatively simple configuration may give rise to a suite of incompatible protocols, propriety data formats and hard-coded behaviours. Thus, a prerequisite to the successful deployment and adoption of AAL is a mechanism for effectively managing this disparity and heterogeneity if an effective mix-and-match approach is to be adopted. To address these issues, the harnessing of an intelligent distributed middleware is proposed.

Embedded agents incorporate a range of attributes that make them a particularly apt solution for realising a distributed software infrastructure for managing embedded sensors. Their autonomous and social capabilities enable them to collaborate in the pursuit of their objectives, which of course will vary in individual cases. In-situ processing of data contributes to flexibility and robustness, and can either replace or complement conventional centralised approaches to data management and decision making. Such an infrastructure provides for extensibility, something that AAL designers need to be conscious of, as design decisions made in initial AAL deployments will affect how the systems evolve over time [1]. For the purposes of this discussion, we consider AAL from a sensor, middleware and interface perspective.

2. The Sensor Layer

In order to meet the individual requirements of the AAL environment this work proposes the use of intelligent distributed middleware implemented on a suite of embedded autonomous agents. As the AAL environment should by its very nature be wide ranging from a topological perspective and in terms of sensing capabilities, a platform that supports interoperability, expandability, reusability and robustness is most useful. The building block employed in the context of this work is the versatile modular Tyndall mote wireless sensor networking system.

2.1. The Modular Tyndall Mote WSN prototyping system

The Tyndall prototyping system has been developed to address a wide array of scenarios in the Wireless Sensor Network (WSN) application space. A highly modular approach to design has been adopted negating the need to replace the mote infrastructure should a change in wireless technology, sensing capabilities or power supply be required [2]. The platform implementation consists of a variable number of layers that are stacked on top of each other in order to satisfy application requirements. There are a number of benefits in adopting this modular approach to node design namely the platform is far more interoperable with a wider range of wireless technologies and should there be a need to change technology the background functionality of the network is retained and the need for sensor recalibration is removed [3, 4]. To date upwards of 40 system layers have been developed including over 25 modular sensor layers. To augment the sensor platform with a smart and intelligent software platform, a Smart Pervasive System Layer has been developed. This comprises a powerful Java compatible processing platform supporting a SQUAWK virtual machine enabling the deployment of certain Java-based agent frameworks. The system is designed to be compatible with the 25mm family of devices available from Tyndall (sensors, transceivers etc) but with a high end processing capability. The envisaged operation and modularity of the system is outlined in the block diagram in Fig. **1 (a).** When coupled with the Smart Pervasive System Layer a number of sensor layers are envisaged as useful tools for gathering contextual information from the AAL environment.

2.1.1 Sensor Layer 1 – Pervasive Monitoring Layer

This sensor layer is built around the various sensor interfaces provided by the Atmel microcontroller in conjunction with a multi-sensor interface layer (ref. Fig. **1 (b)).** This sensor layer is fully compatible with the Tyndall mote 25mm “family” of devices and sensor layers and includes a variety of typical sensors on board which provide useful data within the AAL environment including temperature, humidity, light levels, vibration, orientation and presence. There are, in addition to this standard set of onboard sensors incorporated in the system, a variety of sensor interfaces such as I2C, USART, RS485 and analogue I/O capability for additional “non standard” sensor interfacing.

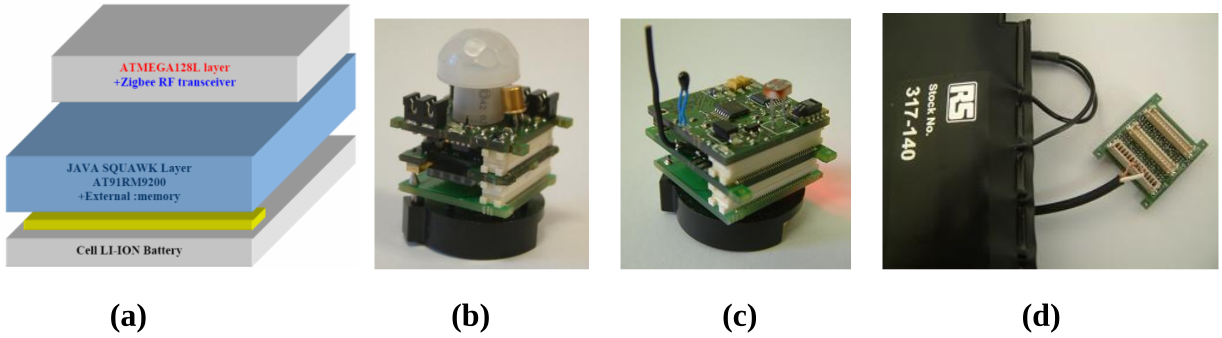


Figure 1. (a) Agent Ready System Block diagram **(b)** 25mm BEM Standard Sensor Layer. **(c)** The Multisensor Layer in the modular stack **(d)** 25mm Generic Sensor Interface Layer connected to a pressure mat.

2.1.2 Sensor Layer 2 - The Multisensor Layer

The Multisensor Layer comprises of a light dependant resistor to measure ambient lighting levels, a thermistor to highlight changes in temperature, relative temperature and humidity sensors, a 3 axis accelerometer to monitor movement and a microphone to detect sound. The Multisensor layer is illustrated in Fig. **1(c)** as part of the modular stack.

2.1.3 Sensor Layer3 - Generic Sensor Interface Layer

The Generic Sensor Interface layer can be interfaced to as many as eight different sensors/devices. In addition to enabling connections, the layer has onboard signal conditioning designed to remove noise. In the context of the AAL platform, this layer has been employed to gather pressure mat readings to detect presence and mobility. The pressure mat configuration is shown below in Fig. **1(d)**.

3. Embedded Agents

Our view is that we need to deploy agents as close as possible to the environment so that much of the decision making and intelligence can occur at run-time and at the leaves rather than the trunk of the sensing and acting infrastructure. The agent-ready platform presented in the previous section enables us to design intelligent and adaptive software systems thanks to the agents’ capability to drive intelligent sensing and acting capabilities in situ. All agents are based on the Agent Factory Micro Edition (AFME) [5], a minimized footprint intelligent agent platform for resource constrained devices. In AFME, as with many other intelligent agent platforms, system functionality is delivered through a combination of imperative and declarative code.

In AFME, the imperative functionality is in the form of a set of perceptors and actuators. The declarative functionality is in the form of commitment rules, which define the conditions under which agents should adopt commitments to perform primitive actions or plans. In short, perceptors generate meta-information (beliefs) about the system state along with information related to the environment, potentially coming from hardware infrared sensors, power monitors, or video cameras. Using this information, the agent will decide, using its internal declarative rule set, on the actions that need to be performed. Actuators provide the functionality for primitive or atomic actions. It should be noted that when an agent adopts a plan or a primitive action fails, a commitment management process is invoked. In AFME, the truth of the belief sentence is evaluated using resolution based reasoning, which is the goal based querying mechanism used within Prolog, as shown in the simple example below:

```
temp_threshold(?threshold)& temperature(?temp)&greaterThen(?temp,?threshold))
>signal(temp_alarm(?temp));
```

In the above example, the *?* symbol represents a variable. The rule states that if the agent holds the beliefs *temp_threshold(?threshold)* and *temperature(?temp)*, and the value of the variable *?temp* (e.g. the temperature in the environment) is greater than the value of the variable ?threshold, the action signal *temp_alarm(?temp)* is activated, causing the agent to send an alarm to its base station.

4. The Middleware Layer

From a Middleware perspective, an Ambient Assisted Living (AAL) system requires many features. From the user's perspective, they need to be able to have new devices automatically integrate into their home environment in a manner consistent with their existing understanding of the system. Furthermore, the middleware must adapt not only to the user's preferences but also to the fact that the user's preferences may change due to an improvement or degradation of their condition. In addition to ensuring the user’s preferences are met, the middleware must also ensure system level performance to meet the requisite Quality of Service. This may mean the opportunistic hibernation of nodes to ensure the network is operational for as long as possible. It is also responsible for adapting the system to failures.

Conveniently, the functionality required of an AAL system will also be present in many other systems which require an intelligent infrastructure of sensors. With this in mind, it is vital that any component developed for this middleware is reusable in many other contexts. For example, a system for the automated tracking of waste disposal, will have many of the same components and thus can reuse elements that have been tested in this context. Reuse will reduce the cost of future systems and reduce testing time. The publish-subscribe bus of our middleware based on the OSGI service oriented architecture allows the application interface to glue together the required system components and provide an appropriate front end to the system.

Finally, the SIXTH middleware [http://www.clarity-centre.org/SIXTH/] also supports the hot swapability of components at runtime as well as the runtime updating of components. This functionality is vital when one element of the system is part of multiple applications. The applications can be updated independently without any effect on neighboring systems. SIXTH provides for the runtime updating of both the embedded and integration components.

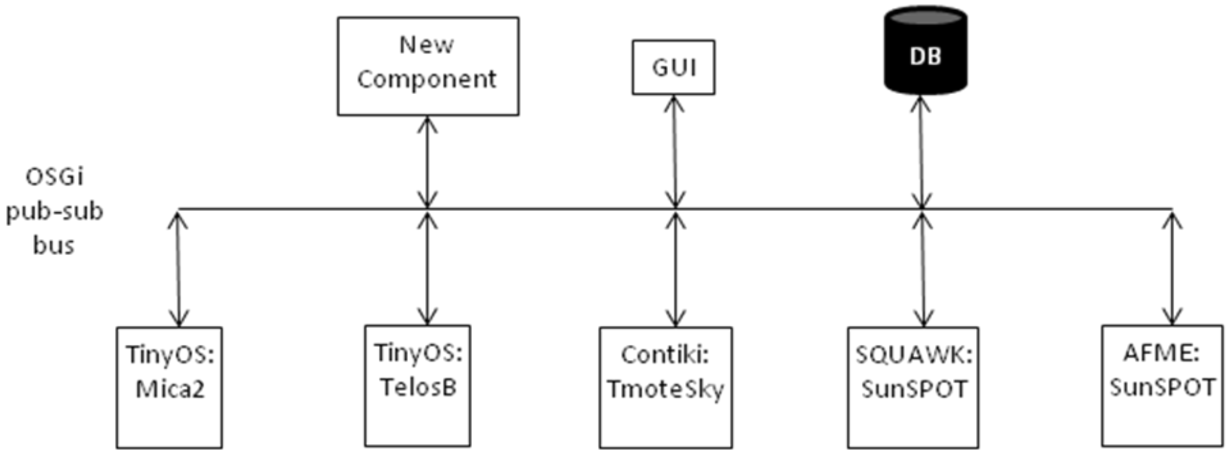


Figure 2. Middleware for Ambient Assisted Living.

5. The Interface Layer

The Interface Layer of an Ambient Assisted Living (AAL) system poses a unique challenge for the design of effective user interfaces. Given the wide scope of the likely user base, developing for a conventional desktop computing environment and assuming a computer literate user-base is not sufficient. Rather we propose supporting multi-modal access to the AAL system, where numerous access devices must be supported, ranging from conventional computing interfaces, to mobile interfaces and the lean-back interfaces of the living-room, such as on the living-room tv. For this AAL system we are developing an initial interface that operates on a living-room TV, which is a social point and an environment in which viewers are relaxed and comfortable to interact with. When developing interfaces for a given device, cognizance must be taken of the inherent device limitations. In our case, on the living-room TV, we considered viewing distance, remote control as the only input device and the requirement of enjoyment-oriented as opposed to task oriented design as the primary limiting factors. Naturally, it is not possible to program a TV, so we are developing in Microsoft XNA and deploying the AAL interface to a Microsoft XBOX 360 home entertainment console. This device natively works with a living room TV and the XBOX controller contains many of the interaction modes that are found on a living-room TV interface. We anticipate integrating other interaction devices at a later date, such as iPads, mobile phones and conventional computers.

6. Conclusions

Diversity of person, situation, environment are characteristic of AAL. Effectively managing this is essential to adoption of AAL technologies. In this paper, the diversity has been explored from a number of aspects and a dynamic agent-based middleware is proposed as a potential solution.

Acknowledgements

This work is supported by Science Foundation Ireland under grant 07/CE/I1147.

References and Notes

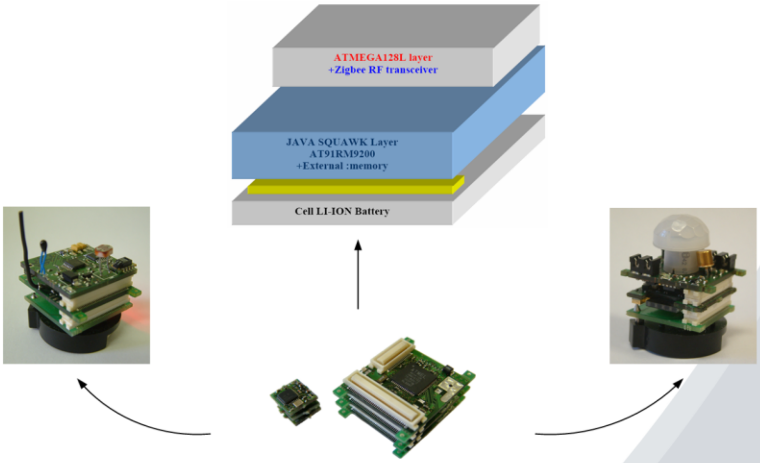
1. O'Grady, M.J.; Muldoon, C.; Dragone, M.; Tynan, R.; O'Hare, G.M.P. Towards Evolutionary Ambient Assisted Living Systems. *Journal of Ambient Intelligence and Humanized Computing* **2010**, 1(1), 15-29. Springer.
2. O’Flynn, B.; Bellis, S.; Mahmood, K.; Morris, M.; Duffy, G.; Delaney, K.; O’Mathuna, C. A 3-D Miniaturised Programmable Transceiver, *Microelectronics International,* **2005**, 1(2), 8-12
3. Menzel, K.; Pesch, D.; O'Flynn, B.; Keane M.; O'Mathuna, C. Towards a Wireless Sensor Platform for Energy Efficient Building Operation, *Tsinghua Science & Technology,* **2008,** 13(1), 381-386.
4. Hautefeuille, M.; O’Mahony, C.; O’Flynn, B.; Khalfi, K.; Peters, F. A MEMS-based Wireless Multisensor Module for Environmental Monitoring, *Microelectronics Reliability*, **2008**, 48(6), 906-910.
5. Muldoon, C.; O Hare, G.M.P.; Collier, R.W.; O Grady, M.J. Agent Factory Micro Edition: A Framework for Ambient Applications. Proceedings of the International Conference on Computational Science (ICCS2006), Reading, UK. **2006**. LNCS 3993, 727-734. Springer.

Managing Diversity in Practical Ambient Assisted Living Ecosystems

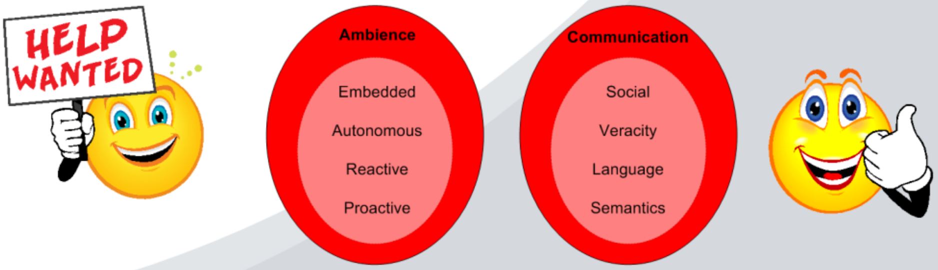
M.J. O'Grady, M. Walsh, R. Tynan, C. Gurrin, M. Dragone, G.M.P. O'Hare,

Ambience demands that AAL technologies are integrated into the home environment in a seamless and transparent fashion. A popular approach is to augment common everyday objects in the home with sensor technologies.

Assistance demands a facility for activity and situation recognition. Only in this way can AAL identify situations that demand intervention. However, the practical implementation of a platform for behaviour recognition is problematic.



A range of sensor platforms and sensor modalities are needed to capture the essence of select aspects of everyday life. Such platforms can be sourced from a range of commercial suppliers. However, the issue of incompatible data formats and propriety protocols is problematic. This results in difficulties for system integration and information fusion activities, rendering the key decision making processes less effective.



Ambience

Embedded
Autonomous
Reactive
Proactive

Communication

Social
Veracity
Language
Semantics

Managing the inherent diversity in AAL demands an adaptable, robust and scalable middleware solution that encapsulates the diverse functionalities and presents a common interface for system integration. Embedded Agents offer intuitive mechanism for realising such a middleware infrastructure.

This work is supported by Science Foundation Ireland under grant 07/CE/I1147.

UNIVERSITY COLLEGE DUBLIN • DUBLIN CITY UNIVERSITY • TYNDALL NATIONAL INSTITUTE