A Middleware for Smart Environments

Crhistian Alberto Noriega Guerra and **Flavio Soares Correa da Silva**[1]

Abstract. A *smart environment* is a context sensitive system based on ubiquitous computing, in which the environment interacts with its inhabitants through embedded dedicated devices. The design and construction of a smart environment requires the collaboration among several areas, such as (1) intelligent man-machine interfaces, (2) pervasive communications, (3) ambient intelligence, (4) scalable systems and (5) mobile computing. It can be very useful if a designer can abstract the required functionalities from these areas to design and build a smart environment. We propose a layer based middleware for mobile devices (mainly focusing on *smart phones*) for the intelligent interaction between devices. Each layer provides specific functionalities and serves as the ground upon which upper - i.e. more abstract - layers are built. We identify the following layers: (1) infrastructure and communications, (2) services and agents, (3) middleware services and (4) collective intelligence. We also propose a formalization of interactions for the specification of services in a smart environment, which is based on *ambient calculus* and also defines the relationships between layers.

1 Introduction

The interaction with computers in daily activities is useful only if computers can be helpful to the users, providing effective means to complete their tasks. These means can be presented as services, which must be consistent with the tasks at hand of users and also adequate considering the available devices for each user.

A smart environment provides to its inhabitants the computational services most adequate for each task at their hands. These services must be adapted according to each task, and can be provided cooperatively by a group of devices whose activities must then be coordinated.

The design of a smart environment can be based on a variety of approaches, as presented in [4], where the required functionalities to build a smart environment have been identified as (1) intelligent man-machine interfaces, (2) pervasive communications, (3) ambient intelligence, (4) scalable systems and (5) mobile computing.

Different approaches provide to the systems designed different middleware systems, frameworks and tools which have as one of their main goals the abstraction of the required functionalities to build a smart environment. For example, we have the Aura project in which task driven computing is the foundational model for interaction between the environment and its devices; the Oxygen project in which several tools have been defined to deal with ubiquitous computing; and projects like GAIA [19], GatorTech, EasyLiving and Smart Media Spaces, in which specific architectures have been proposed to support the required functionalities for smart environments.

[1] University of Sao Paulo, Brazil, email: {cnoriega,fcs}@ime.usp.br. This work has been partially supported by CNPq and Microsoft Research. The authors wish to thank the anonymous reviewers for their suggestions and corrections to the paper.

More recently, different approaches such as the Open Croquet project [10] have implemented a P2P middleware to build virtual worlds which, together with ubiquitous computing result on the union of real and virtual worlds [9].

Our proposed middleware is based on layers [17] to activate the interaction between devices. Each layer provides support to specific functionalities, as follows: (1) infrastructure and communications; (2) agents and services; (3) middleware services; and (4) collective intelligence.

Moreover, each layer abstracts the complexity of implementation of the corresponding concepts.

2 Goals and Motivation

Each individual nowadays carries with him/her a variety of mobile devices, such as smartphones, PDAs, notebooks, portable game consoles such as PlayStation Portable, etc. the mobility of these devices provides individuals with the capability to receive information anywhere and anytime. Following the tenets of ambient intelligence, the interactions with these devices and among devices must be intelligently managed so that it becomes transparent to individuals when performing their tasks.

Interactions must occur naturally and be unobtrusive to the activities of individuals. One way to attend this requirement is to specify interactions based on high-level and flexible descriptions of interactions and services, for example using ontologies to describe coordination protocols between autonomous agents [7].

In the following section we discuss the functionalities required to build a smart environment. In section 4 we describe the functionalities provided by each layer of our proposed middleware and provide examples of smart environments that could be founded on them. Finally, in section 5 we present some preliminary conclusions and proposed future work.

3 Required Functionalities for Smart Environments

Several projects have been introduced in the literature related to the design and construction of smart environments. In [4] we have studied some of these projects and detected the following required functionalities for a smart environment. The projects we have studied can be found in [22, 23, 24, 25, 26, 27, 28]:

1. A computational model to describe the interactions between the environment and its devices.
2. Architectures to support the proposed computational model.
3. A computational model to facilitate the representation of the physical environment as a computational environment.
4. Techniques and applications for ubiquitous computing and communications.

5. Adaptable environments and devices.
6. Intelligent interfaces between humans, devices and the environment.
7. Seamless interaction between any pair of devices [7, 8].

The middleware proposed in this work aims at supporting these requirements, whilst facilitating the design of sophisticated environments through the abstraction of specific functionalities and requirements in different layers of abstraction.

4 Architecture of the Proposed Middleware

In this section we describe the functionalities proposed for each layer in our middleware. We also suggest different projects that could be used to support the implementation of each layer in our middleware.

Each layer defines different entities as first class entities, as follows:

- The infrastructure layer transports messages. The messages include entities of the upper layers.
- The layer of agents and services conveys semantic descriptions of agents and services.
- The layer of middleware services provides pré-built services to convey specific functionalities to the environment.
- The layer of collective intelligence contains more ellaborate entities and artifacts in charge of the coordination of actions involving several devices in the environment.

The interactions between layers is defined using *ambient calculus* [15], so that each entity in a layer is considered na agent in the semantics of ambient calculus.

In figure 1 we have the layers of the middleware and corresponding projects related to each layer.

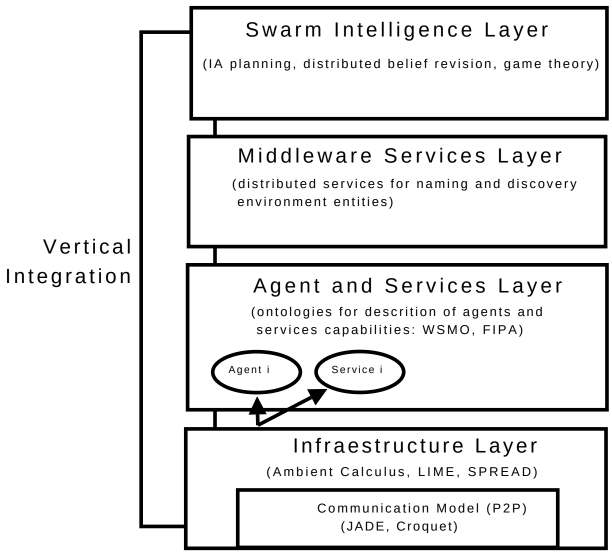

Figure 1. Middleware Layers and Corresponding Projects

4.1 Infrastructure

This layer provides the communications infrastructure for devices. It must support different communications technologies, such as Wi-Fi, Bluetooth and RF, and different architectures, such as P2P and client/server. It is based on the model presented in [20, 21], where messages are placed in a common space visible to the upper layers.

This layer must implement the message exchange infrastructure, which can be based on models such as JADE [11] - which is based on the FIPA model for message exchange in multiagent systems - and Croquet [10] to build collaborative multiuser applications.

4.2 Agents and Services

This layer must contain a repository of agents, services and semantic descriptions of their capabilities. It is responsible for the discovery of the required functionalities implemented in an agent or service that can be found in a device. In [4] we have introduced a mechanism for the discovery of services and agents based on ATMS (assumption-based truth maintenance systems), which matches functionalities with tasks [6]. Moreover, in [13, 14] we find proposals to standardize the descriptions of such functionalities so that other layers in the middleware can retrieve them.

4.3 Middleware Services

This layer contains services provided by the middleware itself and which are distributed across the devices in the environment. Examples of such services are yellow pages and discovery of entities from different layers, management of ontologies and semantic objects.

This layer provides to the other layers resources for:

- Management and maintenance of ontologies, implemented using mechanisms such as e.g. COBRA [3];
- Retrieval and processing of contextual information [2];
- Yellow pages and Discovery of entities based on mobility, as in [18].

4.4 Collective Intelligence

Contains more ellaborate entities (artifacts) to coordinate the entities of the lower layers [7, 8].

In [1, 12] several methods are presented to manage and coordinate services provided by heterogeneous agents. The difference in our case is that the coordination must manage entities of lower layers, i.e. representations of services and agents in different levels of abstraction.

The construction of these entities and artifacts must rely on concepts originating from:

- Game Theory;
- Distributed truth maintenance systems;
- Multiagent coordination;
- AI planning; etc.

5 Conclusions and Future Work

In [5] we find several scenarios of applications of ambient intelligence, with their corresponding challenges related to social and ethical issues, technological issues and industrial issues. As a result, the following topics are suggested as relevant for the construction of ambient intelligence:

1. Specialized hardware development;
2. Open platforms development;
3. Unobtrusive and invisible technologies;
4. Support to personal and social activities; and

5. Safety and reliability.

We intend to build proof-of-concept systems using our middleware, in which these topics are taken into account.

At present, we are working on the specification of the interactions between layers of abstraction, employing *ambient LCC* [15, 16] (Lightweight Coordination Calculus) - which is a variation of ambient calculus - for that. Our goal is to specify the interactions between layers in such way that different implementations of each layer can be built and used in our middleware, provided only that it is aligned with this specification.

REFERENCES

[1] Keith S. Decker and Victor R. Lesser, *Designing a Family of Coordination Algorithms*.

[2] Schmidt A., *Interactive Context-Aware Systems Interacting with Ambient Intelligence*, Ambient Intelligence, IOS Press, 2005.

[3] Chen H., Finin T. and Joshi A., *An Intelligent Broker for Context-Aware Systems*, Adjunct Proceedings of Ubicomp 2003, USA, October, 2003.

[4] Noriega G. Crhistian, *Um modelo para ambientes inteligentes baseado em servicos web semanticos*, Master dissertation, University of Sao Paulo - Brazil, Agosto 2007.

[5] Ducatel K., Bogdanowicz M., Scapolo F., Leijten J. e Burgelman J-C. *Scenarios for Ambient Intelligence in 2010*, *ISTAG*, Febrary 2001.

[6] Wang Z. e Garlan D., *Task-Driven Computing*, Technical Report, School of Computer Science, Carnegie Mellon University, May 2000.

[7] Ramparany, F., Boissier, O., Brouchoud, H, *Cooperating Autonomous Smart Devices*, In: sOc'2003, the Smart Objects Conference, pp. 182-185 (2003).

[8] Naohiko Kohtake, *Smart Device Collaboration for Ubiquitous Computing Environments*, Ubicomp2003.

[9] Jianhua Ma, Laurence T. Yang, Bernady O. Apduhan, Runhe Huang, Leonard Barolli and Mokoto Takizawa, *Towards a Smart World and Ubiquitous Intelligence: A Walkthrough from Smart Things to Smart Hyperspaces and UbicKids*, Pervasive Computing and Communication, IEEE, March 2005.

[10] David A. Smith, Alan Kay, Andreas Raab and David P. Reed, *Croquet - A Collaboration System Architecture*, First Conference on Creating, Connecting and Collaborating through Computing, 2003.

[11] Fabio Luigi Bellifemine, Giovanni Caire and Dominic Greenwood, *Developing Multi-Agent Systems with JADE*, Wiley, April, 2007.

[12] Omicini A., Zambonelli F., Klusch M. and Tolksdorf R., *Coordination of Internet Agents Models, Technologies, and Applications*, Springer, 2001.

[13] Xavier E. and Correa da Silva. F. S., *Expressing Systems Capabilities for Knowledge Coordination*, AAMAS'2002.

[14] Roman D., Lausen H. e Keller U., *Web Service Modeling Ontology (WSMO)*, http://www.wsmo.org/TR/d2/v1.2/, Working Draft D2v1.2, April 2005.

[15] L. Cardelli and A. D. Gordon, *Mobile Ambients*, Foundations of Software Science and Computational Structures, number 1378 in Lecture Notes in Computer Science, pages 140Ű155. Springer-Verlag, 1998.

[16] Sindhu J., Perreau de Pinninck Bas A., Robertson D., Sierra C. and Walton C., *Interaction Model Language Definition*, IJCAI Workshop AOMS Agent Organizations Models and Simulations, 2007.

[17] Sven Apel, Klemens Böhm, *Towards the Development of Ubiquitous Middleware Product Lines*, In ASE'04 SEM Workshop, volume 3437 of LNCS, 2005.

[18] Issarny V., Sacchetti D., Tartanoglu F., Sailhan F., Chibout R., Levy N. and Talamona A. *Developing Ambient Intelligence Systems: A Solution based on Web Services*, In Journal of Automated Software Engineering. Vol 12. 2005.

[19] Renato Cerqueira, Christopher K. Hess, Manuel Román and Roy H. Campbell, *Gaia: A Development Infrastructure for Active Spaces*, In Workshop on Application Models and Programming Tools for Ubiquitous Computing, September 2001, Atlanta, Geogia.

[20] Picco G., Murphy A. and Roman G., *Lime: Linda Meets Mobility*. In Proceedings of the 21stInternational Conference on Software Engineering (ICSE'99), Los Angeles, USA, 1999.

[21] Couderc P. and Banâatre M., *Ambient Computing Applications: an Experience with the SPREAD Approach*, Procs. of the 36th Hawaii Int'l Conf. on System Sciences (HICSS'03), IEEE Comp Soc, 2003.

[22] Steven Shafer, et al, *The New EasyLiving Project at Microsoft Research*, Proceedings of the 1998 DARPA/NIST Smart Spaces Workshop, July 1998.

[23] Project Oxygen, *http://www.oxygen.lcs.mit.edu/*.

[24] Project One.world, *http://www.cs.nyu.edu/rgrimm/one.world/*.

[25] Helal A., Mann W., Elzabadani H., King J., Kaddourah Y. e Jansen E., *Gator Tech Smart House: A Programmable Pervasive Space*, IEEE Computer magazine, March 2005.

[26] Sousa J. P. e Garlan D., *Aura: an Architectural Framework for User Mobility in Ubiquitous Computing Environments*, Proceedings of the 3rd Working IEEE/IFIP Conference on Software Architecture, August 25-31, 2002.

[27] HomeLab, *http://www.research.philips.com/technologies/misc/homelab/*, Philips Research.

[28] Kidd, Cory D., Robert J. Orr, Gregory D. Abowd, Christopher G. Atkeson, Irfan A. Essa, Blair MacIntyre, Elizabeth Mynatt, Thad E. Starner and Wendy Newstetter, *The Aware Home: A Living Laboratory for Ubiquitous Computing Research*, In the Proceedings of the Second International Workshop on Cooperative Buildings - CoBuild'99.