An Embedded Middleware Platform for Pervasive and Immersive Environments for-All

R. Baldoni, C. Di Ciccio, M. Mecella, F. Patrizi, L. Querzoni, G. Santucci – *SAPIENZA Università di Roma, Italy;*
S. Dustdar, F. Li, H.-L.Truong – *Technische Universität Wien, Austria;*
L. Albornos, F. Milagro, P. Antolin Rafael – *Telefónica Investigación y Desarrollo, Spain;*
R. Ayani, K. Rasch – *Kungliga Tekniska högskolan, Sweeden;* M. Garcia Lozano – *Swedish Defence Research Agency, Sweeden;*
M. Aiello, A. Lazovik – *Rijksuniversiteit Groningen, The Netherlands;* A. Denaro, G. Lasala, P. Pucci – *Elsag Datamat, Italy;*
C. Holzner – *Guger Technologies OEG, Austria;* F. Cincotti, F. Aloise – *Istituto Santa Lucia, Italy*

Abstract: **Embedded systems are specialized computers used in larger systems or machines to control equipments such as automobiles, home appliances, communication, control and office machines. Such pervasivity is particularly evident in immersive realities, i.e., scenarios in which invisible embedded systems need to continuously interact with human users, in order to provide continuous sensed information and to react to service requests from the users themselves. The SM4All project investigates an innovative middleware platform for inter-working of smart embedded services in immersive and person-centric environments, through the use of composability and semantic techniques for dynamic service reconfiguration. This is applied to the challenging scenario of private houses and home-care assistance in presence of users with different abilities and needs (e.g., young, able-bodied, aged and disabled). This paper presentes a brief overview of the SM4All system architecture.**

Index Items: **pervasive computing; middleware; wireless sensor networks; composability; ambient assisted living**

I. INTRODUCTION

Embedded systems are specialized computers used in larger systems or machines to control equipments such as automobiles, home appliances, communication, control and office machines. Such pervasivity is particularly evident in immersive realities, i.e., scenarios in which invisible embedded systems need to continuously interact with human users, in order to provide continuous sensed information and to react to service requests from the users themselves. Having the users at the centre poses many new challenges to the current middleware and service technologies for embedded systems, in terms of:

- *dynamicity*: sensors and services are no more static thus the overall distributed system (consisting of all the sensors and devices and appliances) need to continuously adapt on the basis of the user context, habits, etc., by adding/removing/composing on-the-fly basic elements (services offered by sensors/devices/appliances);

- *scalability*: in order to really immerse the users in the system, the number of sensors/devices/appliances could be huge;

- *dependability*: when users are at the centre of the environment, they heavily depend and thrust on the system itself, that should therefore be highly dependable;

- *security and privacy*: if users continuously interact with the system, the security of the overall invisible environment become crucial; therefore, the design of such a system should pay specific attention to privacy preservation, that should be built-in in the system, and not added-on later, as in current design practices.

These considerations require novel techniques and middleware technologies targeted to person-centric embedded systems (i.e., usable in immersive scenarios). Among the most widespread scenarios that require person-centric embedded systems, domotics and home-care assistance are particularly significant in the current technological context. Domotics and home-care are the fields where housing (domus) meets technology in its various forms (informatics, but also robotics, mechanics, ergonomics, and communication) to provide better homes from the point of view of safety and comfort and care for elderly/disabled persons. Traditionally, domotic solutions were provided by one vendor, using one standard for communication, most often closed, and were expensive. This is no longer true. Domotic elements are heterogeneous in all aspects. Devices come from different vendors, have different hardware, network interfaces, and operating systems; they however need to interoperate. Users want to be able to have a unique view on all the hardware entering their home. Therefore, one of today's challenges for the home is total interoperability and cooperation. On the other hand, users wish to seamlessly interact with their home, independently of their status and abilities.

The SM4All (Smart hoMes for ALL) project aims at studying and developing an innovative middleware platform for inter-working of smart embedded services in immersive and person-centric environments, through the use of composability and semantic techniques, in order to guarantee dynamicity, dependability and scalability, while preserving the privacy and security of the platform and its users. This is applied to the challenging scenario of private/home/building in presence of users with different abilities and needs (e.g., young able-bodied, aged and disabled).

In particular, in the SM4All project P2P, service-orientation and context-awareness are merged in novel ways in order to define a general reference architecture for embedded middleware targeted to immersive scenarios, among which the domotics and home-care have been selected as showcases. The

SM4All platform will offer specific support to scalability and dynamicity. By exploiting techniques from autonomic computing (self-* properties) and peer-to-peer systems, the services and the platform will offer such features directly built-in, and not as a subsequent add-on. Finally the SM4All project aims at defining and developing such an embedded pervasive platform for smart houses truly for all, in which users with different abilities and needs (e.g., young able bodied, aged and disabled) can interact with the services provided by the different domotic devices, appliances and sensors through basic and advanced interfaces, being the latter ones based on brain-computer interaction technologies [3]. In such a way, the ambitious target of the SM4All project is to couple, for the first time, advanced research and techniques in embedded and distributed systems with service oriented computing (on the one hand) and accessibility and advanced interaction techniques (on the other hand), thus developing a truly embedded dynamic environment for all.

II.   THE SM4ALL ARCHITECTURE

Clearly, any effort towards this vision requires significant change in the embedded middleware development strategies of the past, with respect to functionality, operational environment, and other aspects. Contrary to current practice, the vision requires that the following characteristics are enforced:

Person-centric awareness: humans are at the center of new immersive environments and all efforts to develop such environments should be initiated and motivated by needs to provide interesting and/or novel experiences to users. Such environments and systems should synthesize all information that is available about each person in a cohesive whole, so that they may offer personalized treatment to individuals or classes of individuals based on their profiles. This requires novel technologies for (*i*) data dissemination, (*ii*) their integration, (*iii*) user profiling, (*iv*) context computation.

Globally distributed, communication-centric & service-centric functionalities: the middleware itself and the provided infrastructural services (e.g., storage and retrieval of service descriptions, communication, etc.) should be managed in a widely distributed manner; therefore, a peer-to-peer paradigm should be enforced, in order to guarantee dynamicity, scalability and dependability.

Openess and maximum-reuse: for economy of scale, reusability, and extensibility, generic embedded middleware should be developed that capture all common aspects of the immersive scenarios. Openess of the middleware is also important. Web service technologies [2], and service-oriented approaches, are a promising solution to this, if adequately complemented with specific solutions for their scalability, dependability, dynamicity and security.

Goal of the SM4All architecture is to seamlessly integrate devices in order to simplify access to services provided by these devices and dynamically compose these services to offer to the end users more complex functionalities and a richer experience of the domotic environment.

Due to the different technologies employed by the devices that are expected to interact within SM4All, the architecture
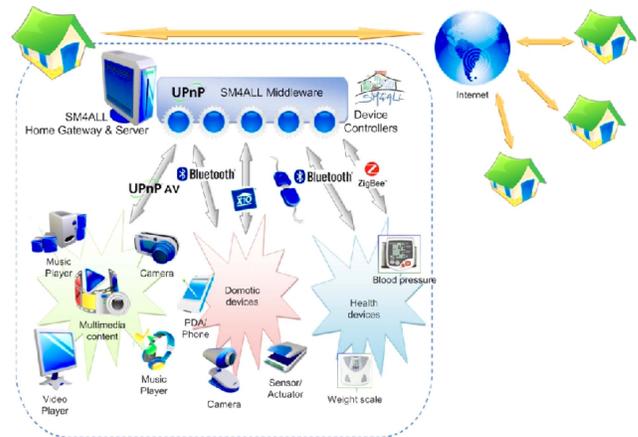

Figure 1: overall view of the SM4All architecture

relies on an abstracting communication layer represented by the UPnP standard [1]. The employment of this abstraction brings significant advantages in the form of: (*i*) seamless compatibility among different communication standards (WiFi, ZigBee, Bluetooth, etc.), (*ii*) discovery of new devices and (*iii*) management of failures or device removals.

Figure 1 gives a high level view of a possible SM4All deployed system, where widely different devices (multimedia, domotic and healthcare devices) interacts using their different communication standards through a UPnP Home Gateway. The gateway represents the interface between the devices the rest of the SM4All middleware.

The SM4All middleware, depicted in Figure 2, is built upon various components organized on three functional layers:

The **Pervasive Layer** consisting of the hardware infrastructure (devices, sensors, appliances) and the embedded software on top of them, i.e., the services, the distributed middleware and the distributed orchestration engine. Such a layer is designed according to the P2P paradigm. The layer also includes components devoted to abstracting the inherent complexity of integrating and offering services provided by widely different devices.

The **Composition Layer** consisting of all the components needed to automatically interpret and satisfy user needs by selecting, composing, adapting, orchestrating existing low level services while taking into account user preferences and the current context of the system. Components located in this layer has the main goal of receiving high level commands issued by users through the interface layer and fulfilling the corresponding complex goals by controlling the execution of lower level services offered by devices deployed within the SM4All architecture

The **User Layer** is devoted to the interaction of the SM4All architecture with final users and administrators. This interaction will be realized through various UIs (traditional computer-based applications, Brain-Computer interfaces (BCIs), WebUIs, etc.).

Components located inside these layers collaborate to provide the complete SM4All experience to users. Here follows a detailed description of the various components:

2
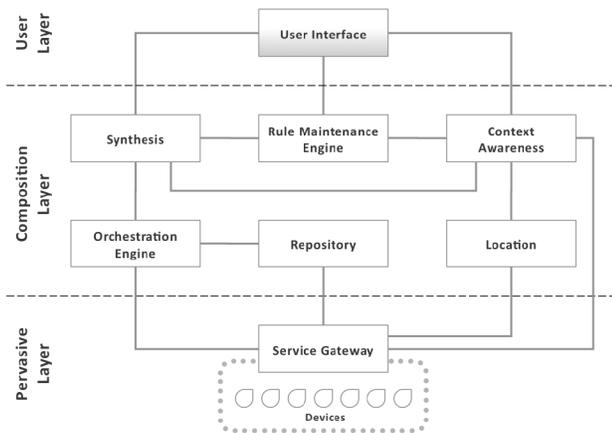
Figure 2: functional blocks in the SM4All middleware

Devices and the Service Gateway. Sensors are devices for the measurement of physical quantities. There is an ever increasing variety of sensor types, ranging from simple thermometers to self-calibrating satellite-carried radiometers. In SM4Allforall a sensor is the utmost tentacle tip of the system, it measures changes and states in the human carrier as well as the surrounding device and environment while reporting the data to the other components. Similar to sensors, the environment is full of other different devices (e.g., typical home appliances) which are able to act on the environment itself and to change its state (e.g., an appliance for opening the window can change the luminosity value sensed by a sensor). Both sensors and devices export their functionalities as *embedded services*. In order to be dynamically configured and composed, embedded services need to expose semantically rich service descriptions, consisting not only of (*i*) interface specifications, but also of (*ii*) specifications of the externally visible behaviors, of (*iii*) offered QoS, and of (*iv*) security requirements. The Service Gateway is the component responsible for letting embedded services interact with the upper layer components of the architecture by publishing available service descriptors, executing service instances, decoupling interactions from the specific communication mediums, etc. This component realizes an abstraction layer wrapping all of the devices currently in the house.

Synthesis Component, Receives high level goals as input from other components and outputs more fine grained execution plans. The main purpose of this component is to translate a high level goal, that express complex actions, in a plan constituted by simple tasks that can be easily assigned to physical or logical devices for the execution [4]. Goals can be received as inputs both from the User layer and from the Rule Maintenance Engine. The translation from goals to plan is executing taking into account information obtained from the Context awareness component.

Orchestration Engine. Receives detailed plans from the Synthesis component and consequently orchestrates a set of available services, so that plans may be actually executed. The orchestration of available services is realized by interacting with the Repository component through which service descriptions can be retrieved.

Repository. Represents a general repository for service descriptors, ontologies or other semantic data.

Rule Maintenance Engine. Manages automatic actions for the system to fire when predetermined conditions hold. Rules can be defined by administrators through the User Layer. The triggering of rules to be applied, since they are automatic, is caused by changes in the current system and is therefore indirectly caused by the Context Awareness component when the predefined conditions hold. Its output is the request to fulfill a goal directed to the Synthesis component.

Context Awareness. This component is expected to collect raw data from devices, process it and store the processing results in order to provide other blocks with an up-to-date picture of the current system status. This status constitutes the context. Also user preferences and profiles, received from User layer components, are stored as part of the context.

Location. a very important kind of information that constitutes the context is represented by the location of objects and people within the environment where the SM4All middleware is deployed and running. Due to the complex mechanisms involved in extracting location information from various entities a specific component is devoted to this duty.

Both the Context Awareness and Location components interact with the Pervasive layer in order to obtain raw data from physical devices.

User Interface. This is the component devoted to capturing and elaborating user inputs, so that they can become invocations for components in the lower layers. This module offers multiple interfaces. These interfaces will be used by users to request new goals to be fulfilled, new rules to be recorded or new preferences to be set up.

III. CONCLUSIONS

The aim of the SM4All project is to provide users of domotic services with a semantically enriched environment where raw services exposed by widely different devices can be seamlessly integrated, composed orchestrated and invoked in order to provide more complex services. These services must be furnished to the user while guaranteeing their reliability and their performance. The SM4All architecture and its middleware introduced in this paper will constitute the fundamental block upon which the final SM4All experience will be built.

REFERENCES

[1] UPnP forum. `http://www.upnp.org/`.

[2] Web services architecture, w3c. `http://www.w3.org/`.

[3] F. Aloise, S. Marchionni, G. Romito, AM. Brouwer, Jan B.F. van Erp, D. Mattia, F. Babiloni, S. Salinari, M.G. Marciani, and F. Cincotti. P300-based brain computer interface: to operate domotic appliance. Proceeding BCI Workshop, Graz 18-19 September, 2008.

[4] G. De Giacomo, M. Mecella: Automatic Web Service Composition. Tutorial IEEE ICWS/SCC 2006.