VECIMS 2007 – IEEE International Conference on
Virtual Environments, Human-Computer Interfaces, and Measurement Systems
Ostuni – Italy , 25-27 June 2007

HI³ Project: Design and Implementation of the Lower Level Layers

G. Varela, A. Paz-López, S. Vázquez-Rodríguez, R. J. Duro
Integrated Group for Engineering Research
Universidade da Coruña
Spain

Abstract *–We are interested in the development of human-centered and ubiquitous technologies in social environments. In this line, and in the framework of a global software architecture (HI³) for this type of applications, the paper is devoted to the presentation of the work carried out for the design and implementation of the layer that is more closely linked to the hardware. It is in charge of communicating with the physical layer and it is responsible for the abstraction of the field elements. Special attention has been paid to the coherence with the philosophy, design premises and functionalities of the whole system. We have also determined a hardware configuration that, integrating standards where possible, is better adapted to the requirements of the architecture. The elements introduced here were validated on a real implementation of the system.*

Keywords *– Ambient Intelligence, ubiquitous computing, distributed sensing/actuation.*

I. INTRODUCTION

There are currently many hardware and software technologies competing for the domestic and social market. Each one of them tries to provide optimal solutions for its portion of the market. The user is immersed in a technological world that requires ever increasing participation, knowledge and training in order to benefit from it. Some visionaries imagined a very different scenario [1] in which technology, present in every aspect of life, would be designed in order to serve the individual who would be able to extract maximum performance from it in a natural and intuitive manner. Currently there are many research groups working on this idea, which has been called pervasive computing, ubiquitous computing or ambience intelligence.

Some researchers work from a global perspective, as in the case of the Oxygen project at the MIT Computer Science and Artificial Intelligence Laboratory [2][3]. It enables pervasive, human-centered computing through a combination of specific system and user technologies that directly address human needs. In the Ambient project, at IPSI: Institut Integrierte Publikations und Informationssysteme [4], efforts are concentrated on the design of artifacts that, integrated in the working environment, permit new ways of interacting with the environment. In other cases, such as the Aware Home Project at the Georgia Institute of Technology or the Intelligent Inhabited Environments Group at the University of Essex, research is constrained to domestic environments [7]. A special mention must be made of those projects for improving health or personal assistance aimed at people with special needs, such as the Proactive Health Research Project, which is a collaborative effort between Intel and researchers at a number of universities.

We are interested in the development of human-centered and ubiquitous technologies in social environments. The environment for a person must be understood as an extended environment. In it the person works, has its family life, its recreational time and its social life. Therefore this environment involves the workplace, the home, car, or public places such as malls or sports centers, among others.

Some research groups have created experimental environments in which to try out new human-centered interaction technologies, such as the iRoom at Stanford University Computer Science Dpt. [8] or the Intelligent Room [9] at MIT Computer Science and Artificial Intelligence Laboratory. Other projects like PRIMA, hosted by the Institut National de Recherche en Informatique et en Automatique, concern technologies for the perception and recognition of human action for intelligent environments and man-machine interaction.

The work presented in this paper was carried out in the framework of a broader multidisciplinary Project defined by our group and which we call HI³ technology. Its main objective: to create Humanized, Intelligent, Interactive and Integrated environments.

In a Humanized environment, people do not need to undergo training in order to make optimal use of technologies; they are transparent to the user. Consequently, to a large extent, the users are freed from their responsibility of explicitly interacting with technology. It is the environment the one that, in an active way, extracts information from the user and its context. An Intelligent environment must exhibit proactive, adaptive, predictive and autonomous behavior. It must be capable of learning from the use and habits of the users in order to predict and anticipate them. In this Interactive scenario, natural and intuitive interfaces must be developed for people to interact with each other and the environment itself. Finally, in an Integrated environment, the different technologies complement each other in a way that is transparent to the user.

In addition, HI³ technology must meet some design premises such as connectivity, determinism, fault tolerance, modularity, scalability, security, real time operation and ease of maintenance. To achieve all of this, the HI³ software architecture has been conceived as a layered agent based architecture [2][3]. This favors better abstraction and distribution of the functionalities of the system and at the same time decreases the coupling between them.

From this point of view and in terms of the Integration property, in this paper we describe, on one hand, the work carried out for the design and implementation of the layer that is more closely linked to the hardware. It is in charge of talking to the physical layer and is responsible for the abstraction of the field elements. Special attention has been paid to the coherence with the philosophy and design premises and functionalities of the whole system. On the other, we have determined the hardware configuration that, integrating standards where possible, is better adapted to the requirements of the architecture.

From this point of view, the proliferation of incompatible field technologies has led to the generation of standardization projects for field sensing and actuation. One of these is the IEEE-1451 standard [5][6]. However, in social environments, the solutions in this line are scarce [4], in general they are based on proprietary field buses. To achieve our objectives we have taken inspiration from the IEEE-1451 project but, as it is a standard with an objective that is different and more general than that of HI^3, we have adapted it to our context and based on it we have developed some hardware devices, as well as software components, as a base for its implementation.

The paper is organized as follows. In a first section we present the global structure of the HI^3 architecture in order to provide the context for this work. We then describe the model proposed, starting from an analysis of the IEEE-1451 standard. This model also includes the solutions adopted for the software abstraction of the physical layer. The next section is devoted to the experimental environment. It describes the hardware on which this model was implemented, including some devices designed and implemented directly for it, and interconnected through an Ethernet network, as well as some commercial devices corresponding to other types of field buses. All of them operate transparently under the HI^3 architecture.

II. STRUCTURE OF THE HI^3 ARCHITECTURE

Figure 1 displays the structure of the layered software solution adopted for the HI^3 architecture. The communication between layers is vertical so that a given layer can only communicate with the one right on top or below it. Within a layer, horizontal communications may also take place among the elements that make it up so as to obtain the functionalities of the layer.

The Device Access Layer is divided into two sublayers:

- Device Controllers. It is a conceptual layer that contains the controllers (both, proprietary and open) for the different devices connected to the hardware communications bus.
- Device Access API. It defines an interface so as to homogeneously access devices and abstracts the rest of the elements from particular implementations for the management of the devices. The elements of this layer play the role of an adapter.

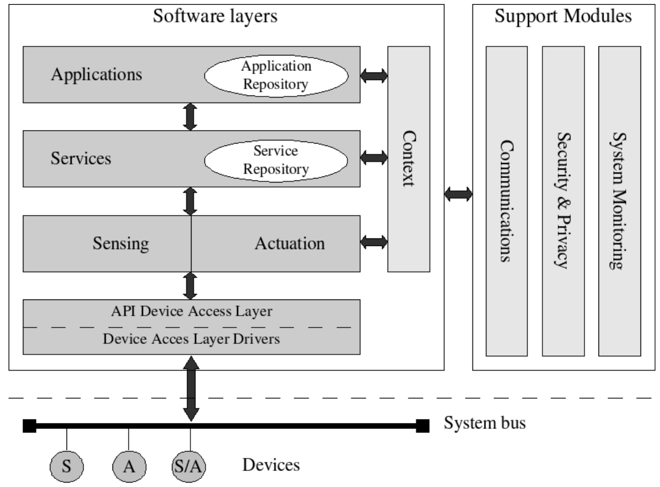

Figure 1 Global architecture for HI^3

The Sensing and Actuation layer contains two clearly different types of elements: sensing elements and actuation elements. The sensing elements are in charge of accessing, through the Device Access API, the information provided by the “sensor” type devices. In the same way, the actuation elements must provide, again through the Device Access API, the methods required in order to command the actuation of the “actuator” type devices.

The service layer is made up mainly of elements we will denote as “services”. A service is defined as an element designed for solving a particular high level task and which, by itself, does not provide a complete and final solution for a use case the system is expected to solve. Services may be compounded so that a given service may use other services registered in the system.

The Applications layer is the highest level layer and the one that hosts the elements representing and implementing applications that provide particular functionalities a user expects from the system. These applications make use of services that are registered in the system in order to carry out their tasks.

The Context is a common access component for the three higher level layers of the architecture. Its objective is the exchange and management of information generated throughout the system. One of its main functionalities is that it represents the current state of the environment.

Finally, this architecture defines a set of support modules encompassing those components that provide basic functionalities for the correct operation of the whole system. Among them we must point out the modules devoted to handling all the aspects related with communications, security and privacy or the continuous monitoring of the system itself.

This paper concentrates on the design and implementation of the Device Access Layer as well as its deployment in an experimental distributed sensing environment that integrates different technologies.

In the framework of the application of this architecture we find that there are many sensing and actuation devices used as a connection between the intelligent higher level computational models and the physical environment over which these models operate. They give rise to a series of problems to be solved by a heterogeneous and distributed device access model that purports to be as generic and functional as possible:

- Difficulty of integrating and interoperating groups of devices of very different natures.
- Portability of the software components to different hardware platforms.
- Risk of translating the inherent complexity of the hardware environment to higher levels of the software architecture.
- Risks derived from the excessive dependence on a given technology or hardware vendor.
- In a real environment there may already be all sorts of hardware elements present, such as classical domotics buses, which require their integration within the architecture.

The device Access model proposed in this paper is aimed at solving these problems, providing a general solution that abstracts the devices and distributed sensing networks from the rest of the architecture. To this end a lot of emphasis is placed on the definition of a standard model that facilitates the integration of the different technologies. This model will be based on the philosophy proposed by the IEEE1451 family of standards.

In addition, a low cost hardware prototype that permits connecting very different sensing/actuation devices to an Ethernet type bus has been implemented in this work. This element, together with the software model that supports the whole system, allows for the fast, easy and cheap deployment of a distributed sensing/actuation network that operates over an Ethernet communications bus.

III. MODEL PROPOSED

The model for accessing and managing devices proposed here attempts to conform a solution that solves the problems described above following the principles proposed in the general HI^3 model: application generality, scalability, modularity, transparent incorporation of different technologies, adaptability, fault tolerance, etc. All of this is achieved in the framework of a distributed, adaptive, proactive, and highly autonomous paradigm.

The base for this proposal is the IEEE1451 family of standards. This family was generated with the objective of addressing the problems of distributed sensing as a whole. To this end, this family is divided into different standards, each one of them aimed at solving a particular set of problems.

- The IEEE1451.2 and higher standards (IEEE1451.X from now on), concentrate on the definition of standardized physical interfaces for connecting sensors and actuators to microcontrollers.
- IEEE1451.1, on the other hand, is devoted to the problem of developing sensing applications. It proposes a software model that abstracts the typical features of a sensing/actuation node in a network.
- IEEE1451.0 is in charge of filling in the gap between the .1 and .X standards, defining an API that allows applications developed over, IEEE1451.1 to transparently use the IEEE1451.X standards.

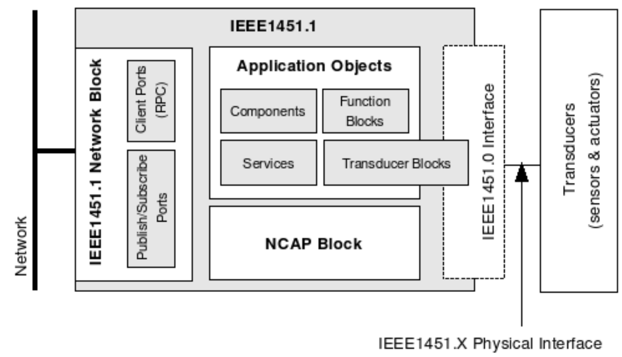

Figure 2 Block diagram of an IEEE1451 node

IEEE1451 proposes, above all, a very powerful and versatile philosophy for developing sensing/actuation applications. It is appropriate for any kind of remote sensing/actuation applications and it is based on the ideas provided by the IEEE1451.1 standard.

The IEEE1451.1, *Network Capable Application Processor (NCAP) Information Model*, standard has the objective of defining a software object model that may be used as an abstraction of the characteristics that a sensing/actuation node should have. This way, it leads to the definition of a series of software components (see Figure 2), which, distributed over three layers (network, application and sensing/actuation devices), constitute a framework for the development of any type of application that requires distributed sensing/actuation capabilities.

This framework provides a large degree of versatility in the development of applications. However, these same characteristic introduces several consequences in the final implementations that are not completely appropriate for the domain we are interested in:

- Such a generic view of sensing/actuation leads to a very high level of complexity and this makes its implementation in hardware systems with limited computational resources difficult.
- As it is a very generic paradigm, it leaves to the vendors the responsibility of defining the network protocols, leading to possible interoperability issues among products from different vendors.

This, together with the fact that there are very few implementations, justifies the need for developing a particular version of IEEE1451.1 adapted to the requirements of the HI^3 architecture. Consequently, some of the concepts described in the standard are left out, but the network layer and some of the proposals for the application utilities layer, are preserved without using their whole implementations (*Function Block,* *Services, Components, etc.)*. In addition, due to the fact that the market is saturated by many proprietary sensing/actuation systems, we think it is more adequate to provide a proposal that tends towards the integration of different hardware sensing/actuation technologies.

Therefore, the solution proposed here bases some of its concepts on ideas of the applications layer proposed by IEEE1451.1, supported by a software library with the network services defined within the standard. They are based on a distributed object system which provides, on one hand, remote procedure invocation services and on the other a publisher/subscriber type service that permits multicast communications among the different components.

In order to cover all the requirements commented in the description of the HI^3 architecture, a distributed model (see Figure 3) based on the network and application services adapted from IEEE1451.1 is considered. This model is supported by three basic pillars:

- Unified and generic view of the different hardware devices available in an intelligent environment.
- Definition of a paradigm for the operating model of a domestic and social sensing/actuation network.
- Abstraction of the device network.

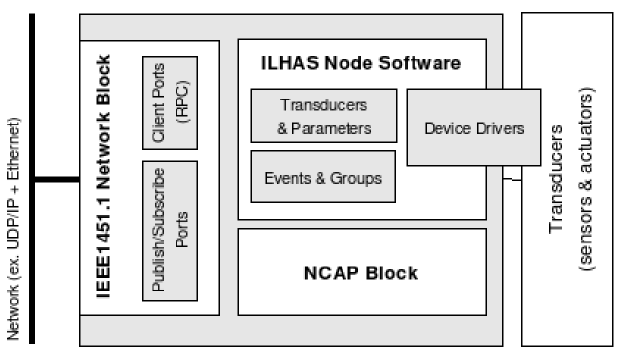

Figure 3 Block diagram of a node in the proposed model

As a first requirement, it is necessary to abstract the vision the system would have of the physical devices, providing a higher level description of them.

This vision is based on the concept of parameter, which can be defined as the properties a given sensor or actuator is capable of reading and/or modifying. These properties are usually values related with the physical environment that surrounds the system and which allow it to know the state of the environment as well as to modify it.

From this abstraction a representation is obtained for the devices that is independent from the particular characteristics of the hardware used to implement them. If this vision is supported by a well known and preestablished parameter hierarchy and a device driver model, a software solution may be obtained that facilitates the integration of devices from different makers operating over heterogeneous technologies. Consequently, the higher level layers of the architecture will only need to know what actuation/sensing capabilities are available and not their specific implementation.

Additionally, using this approach we have a framework that is sufficiently generic and versatile so as to facilitate the development of new sensing elements which, once integrated in this parameter hierarchy will immediately be available to the applications that have been developed.

As a complement to the unification of the vision of the devices it is also necessary to provide an abstraction of the behavior of the device network. This behavior paradigm must be adapted to the requirements of the HI^3 architecture and at the same time must facilitate the integration of the various preexisting technologies.

To address this issue we have considered an approach based on events, parameter read/write operations and their grouping (see Figure 4). Thus, the model we propose allows the higher level layers of the architecture to define different types of events over the various parameters, producing an asynchronous operation in which the software components that are interested in a given parameter may be notified about the changes that take place.

This asynchronous behavior is complemented with the possibility of reading and writing the parameters that are available in a given device as well as the capability of acting over multiple parameters in different devices simultaneously. This is achieved by making use of the multicast communication functionalities provided by the implementation of the IEEE1451.1 based network layer we have developed.

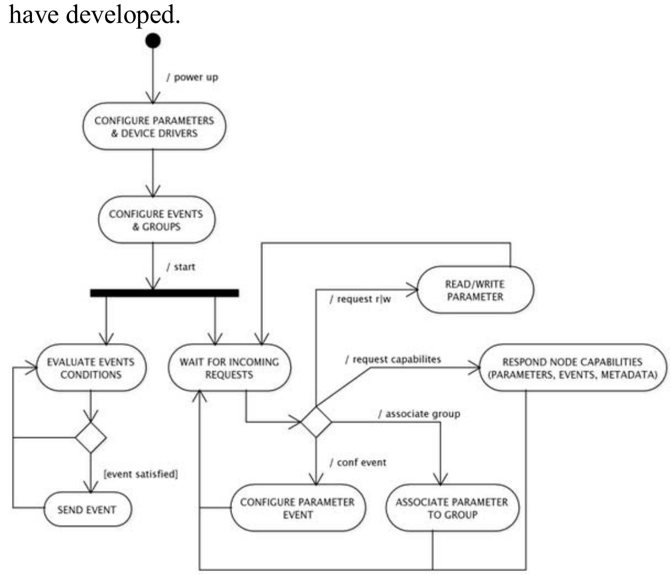

Figure 4 Behavior of the system

Thus, the sensing layer of our system may indicate that some devices must aggregate some of their parameters in a given group and later send write requests over it. As a result, the system will be able to act over multiple devices and areas of the environment in a simple and simultaneous way.

The device network behavior model must be enhanced with a logical vision of it that frees the sensing/actuation stage of the architecture from having to handle the particularities associated with the physical distribution of the devices. As a result, an abstraction of both, the topology and the location of the networks is achieved. To this end, the model facilitates the inclusion of nodes that can act as bridges between different sensing/actuation networks hiding the particularities of each technology.

These facilities, together with the abstractions and paradigms presented previously, provide the higher levels of the HI^3 architecture with a homogeneous vision of the complete underlying hardware structures.

The idea that summarizes the whole philosophy of the model designed here is reflected in figure 5. This figure displays two visions of a complex and distributed scenario that is made up of different communications buses and different types of devices with diverse features. The vision from the hardware point of view is the real and physical vision of the environment. The vision from the software point of view corresponds to the homogeneous and abstract vision that a software component belonging to the HI^3 system would have by incorporating the device management model proposed here.

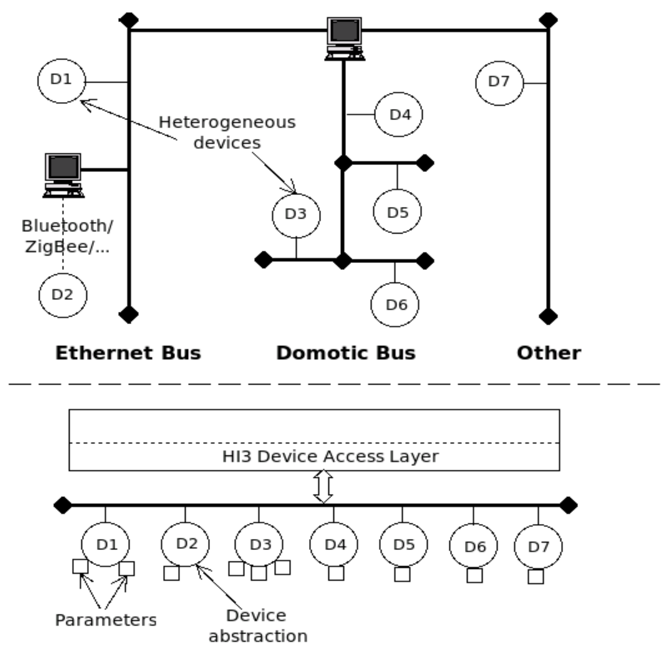

Figure 5 Physical view vs. Logical view of the hardware architecture.

IV. IMPLEMENTATION

Software implementations of the model proposed

Two software implementations of the model proposed in this paper have been obtained. One of them uses the JAVA language, thus benefiting from the portability of this platform. This implementation may be deployed in very different hardware platforms such as PCs, mobile devices or nodes that act as bridges to other technologies.

In addition, an alternative implementation in the C++ language was developed. It is aimed at its use in embedded devices that are characterized by limited computational resources.

These two implementations of the model follow the same specification of the network protocol to implement the IEEE1451.1 network layer used in the system, and, consequently, their integration and interoperability is guaranteed.

Distributed sensing/actuation hardware node

The buses and commercial devices aimed at the home environment are characterized by their high cost and the use of proprietary technologies. As a consequence, we have designed and implemented a hardware device (see Figure 6) that would act as distributed sensing/actuation node. This device has been designed to be used over a communications bus such as Ethernet as this is currently the de facto standard for other types of applications.

Figure 6 displays a prototype of this device. In its implementation we have contemplated widely used hardware components (such as Atmel AVR microcontrollers) as well as open-source software, both for the operating system as well as for the development tools. It has 64 digital and 20 analog inputs/outputs, as well as a 10Mbit network interface. This is a laboratory prototype which, in this implementation, has several sensors and actuators connected to it. The final implementations of the model would be even smaller, as generally they would not require as many inputs/outputs or any of the monitoring elements we have introduced for tests, leading to the possibility of deploying a low cost sensing/actuation network that operates over open technologies.

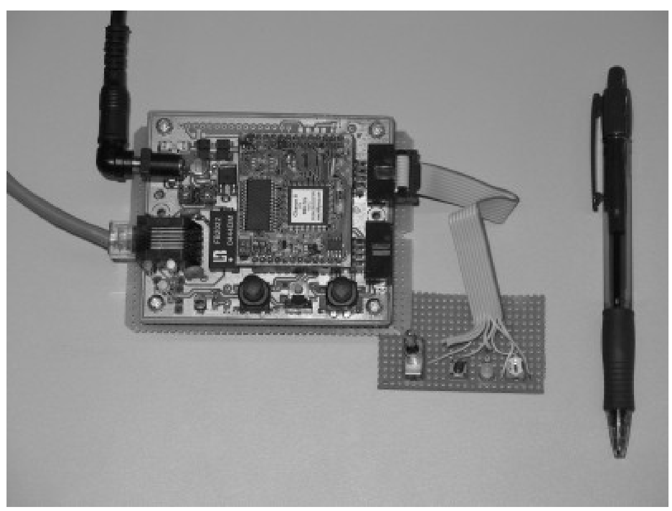

Figure 6 Hardware prototype of a sensing/actuation node

Validation

The validation of the model proposed was carried out on an experimental environment that includes a commercial domotic bus (EIB/KNX) as well as the nodes developed for an Ethernet network based bus. This implementation of the environment was carried out in a room that has numerous types of sensors and actuators related to illumination, blinds, temperature, presence, etc.

The C++ implementation of the architecture was used for the nodes developed and the JAVA version was considered for the construction of a bridge node to EIB/KNX buses. The results of the different tests led to correct performance of the whole system and demonstrated that the higher levels of the architecture operated perfectly with the abstracted vision of the hardware devices and did not need to have any idea of what was over which technology, they just saw different parameters provided by abstract hardware. Thus, an integration of the two technologies is achieved obtaining a homogeneous vision of the device network as was the objective of this work.

V. CONCLUSIONS

We have seen that in a Humanized, Intelligent, Integrated and Interactive environment, such as the one that was defined here, there are a set of problems inherent to the management and integration of physical sensing/actuation device networks. A model that defines a general solution in these types of scenarios was proposed and implemented. Additionally, we have seen that the solution proposed provides a series of desirable features in terms of being a standard model in the management of distributed sensing/actuation networks:

- Scalability. It is a feature that inherent to a P2P type architecture, which is reinforced by the inclusion of one more abstraction level in the sensing/actuation layer of the global HI^3 architecture.
- It does not depend on a particular network technology for communications among devices. The model proposed permits using different base network technologies.
- It provides an abstraction of the hardware devices in two levels.
    - Providing a homogeneous vision of the devices.
    - Providing a generic representation of the information handled by the devices.
- It facilitates a model for the interoperation of different device networks with different bus technologies, different physical locations, etc.
- Incorporating new devices is simple and follows a standard procedure.

ACKNOWLEDGEMENTS

This work was partially funded by the Ministerio de Educación y Ciencia of Spain through projects DEP2006-56158-C03-02 and DPI2006-15346-C03-01 and Xunta de Galicia DXID under project PGIDIT06TIC025E.

REFERENCES

[1] M. Dertouzos. The Unfinished Revolution: Human-Centered Computers and What They Can Do For Us, Collins, 2001.
[2] Michael H. Coen. Building Brains for Rooms: Designing Distributed Software Agents. In Proceedings of the Ninth Conference on Innovative Applications of Artificial Intelligence. Providence, Rhode Island, 1997.
[3] Nicholas Hanssens, Ajay Kulkarni, Rattapoom Tuchinda, and Tyler Horton. Building Agent-Based Intelligent Workspaces. In ABA Conference Proceedings. June 2002.
[4] C. Magerkurth, R. Etter, M. Janse, J. Kela, O. Kocsis, F. Ramparany. An Intelligent User Service Architecture or Networked Home Environments. In Proceedings of 2nd International Conference on Intelligent Environments, Athens, Greece, July 5-6, 2006. pp. 361-370.
[5] IEEE Standard for Smart Transducer Interface for Sensors and Actuators-Network Capable Application Processor (NCAP) Information Model, IEEE Std 1451.1-1999.
[6] A Standardized Approach for Transducer Interfacing: Implementing IEEE-P1451 Smart Transducer Interface Draft Standards, National Institute of Standards and Technology.
[7] 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 Position paper, October 1999.
[8] Brad Johanson, Armando Fox, Terry Winograd. The Interactive Workspaces Project: Experiences with Ubiquitous Computing Rooms. IEEE Pervasive Computing Magazine 1(2), April-June 2002
[9] Michael Coen. Design Principles for Intelligent Environments. Proceedings of AAAI'98, 1998.