Journal of Communication and Computer 10 (2013) 1529-1553

A Privacy Taxonomy for the Management of Ubiquitous Environments

Valderi Reis Quietinho Leithardt[1, 2], Guilherme Antonio. Borges[2], Anubis Graciela de Morais Rossetto[2], Carlos Oberdan Rolim[2], Claudio Fernando Resin Geyer[2], Luiz Henrique Andrade Correia[3], David Nunes[4] and Jorge Sa Silva[4]

1. Institute of Technology, National Service of Industrial Training (SENAI), Porto Alegre 91140-000, Brazil

2. Institute of Informatics, Group of Parallel and Distributed Processing (GPPD), Federal University of Rio Grande do Sul (UFRGS), Porto Alegre 91509-900, Brazil

3. Federal University of Lavras, Lavras 37200-000, Brazil

4. Coimbra University, Coimbra 3030-290, Portugal

Received: October 10, 2013 / Accepted: November 15, 2013 / Published: December 31, 2013.

Abstract: Pervasive and ubiquitous environments must handle the detection and management of users, devices and services, while guaranteeing the privacy of both the users and the environment itself. Current techniques for handling privacy found in the literature treating the subject in various ways, while concentrating on the device management, communication protocols, user profiles and environmental access. This paper examines a control model for privacy in pervasive environments from the perspective of the environment. A prototype was devised and tested to validate the generic model of privacy which was also used to compare taxonomic concepts in the literature. Moreover, the prototype was devised and tested to validate the generic model of privacy for control and manage various users, devices and environments and so on. The prototype was based on Percontrol (a system for pervasive user management), which was only intended to identify users using Wi-Fi, and now it is capable of managing temperature, luminosity and other preferences, measured by a WSN (wireless sensor network) embedded to Percontrol, and the data treatment is done by an ANN (artificial neural network). Results confirmed the viability of device detection with Wi-Fi, Bluetooth and RFID (radio frequency identification) for an increases slight of the latency in registering new devices on the system.

Key words: Percontrol, generic privacy model, ubiquitous environment, artificial neural network.

1. Introduction

In a pervasive environment, computational resources are come omnipresent in people's daily lives and are all interconnected with the objective of providing accurate information and services, regardless of the time or place [1]. Thus, the environment is filled with computational devices that are ingrained in such a way that using them becomes "second-nature", and thus creates the illusion that they just "disappear" and become a normal part of people's daily lives. In recent years, a new computational paradigm is emerging with the emergence of mobile devices, for which the calculation is highly dynamic and must adapt fast to environmental changes. This phenomenon is caused by the user's own mobility and in situations where the processing power exists within small multi-function mobile devices such as mobile phones, smartphones and PDAs (personal digital assistants) [2].

According to Ref. [3], ubiquitous computation is the kind of computation that makes life simpler; digital environments can sense and process information, by being adaptable and becoming

Corresponding author: Valderi Reis Quietinho Leithardt, Ph.D. candidate, research fields: privacy and ubiquitous computing. E-mail: valderi.quietinho@inf.ufrgs.br.

pro-active towards human needs.

Weiser [1] forecast new systems and environments that would be full of computational resources capable of providing services and information whenever necessary ("everywhere, every time computing"). Thus, he proposed a continuous integration between the environment and technology, with the aim of helping people carry out their everyday activities within this environment [1].

The current trend in computing is the usage of "invisible" computers, where the man-machine interaction is governed by non-traditional means; instead of the traditional keyboard and mouse, touchscreen and motion controllers are quickly becoming the standard input mechanisms. Computers are now set up with the aim of responding to user-stimulus, without the need for direct user interaction. This concept is close to the idea of Pervasive computing, since machines are distributed within the environment in a non-perceptible fashion; through the use of sensors or other means of communication such as RFID [4], Bluetooth or WSNs (wireless sensor networks) [5]. These machines communicate between themselves, the users and the environment, and their tasks are modeled to suit everyone's needs in a better way. Pervasive computation should be context-aware and intelligently adapted to finding better solutions to the most diverse situations.

Pervasive computation deals with many situations that have no equivalent in traditional computation: Common among these are changes in the presence of users, location conditions, service availability (such as weather forecasts or clock synchronization), and computational context. Privacy plays an important role in these situations, since a user might not want to be located or share his data during a certain time. These requirements should be met by the pervasive environment, by reducing the processing of unnecessary data and increasing the overall security level and the performance of service management tasks.

We propose a privacy control model for pervasive/ubiquitous environments to properly address the requirements of this pervasive computation; this handles as many requirements related to the environment as possible. In the current literature, several scientific papers were found which adopted an approach to privacy control that involved using different techniques that focused on the user himself or the devices, communication services and privileges which he is able to access.

However, we consider a broad range of different scenarios present in the real world (e.g. churches, libraries and football stadiums), although no one specific work addresses all the necessary requirements that can be found in all the existing scenarios.

The rules and regulations that govern each one of us, are mostly determined by the context of our situation, and the context is closely tied to the environment where we currently reside. Thus, most privacy control mechanisms that focus on the users, devices, communications or services usually lose their validity when removed from the environment for which they were designed.

The main contribution made by this work is to outline a novel model for privacy that is focused on the pervasive/ubiquitous environment, and seeks to bring the concept of Pervasive computing closer to the real world. However, solutions for the problem of security in pervasive/ubiquitous computation will not be addressed, such as techniques for avoiding attacks or encryption algorithms; nor will solutions for restricting users and/or devices, or the services and communication mechanisms available to them.

This paper is structured as follows. In Section 2, an analysis of the state of the art in the literature is conducted, and some key concepts of privacy in pervasive and ubiquitous environments are defined and discussed. A taxonomy for privacy in ubiquitous environments is outlined in the Section 2.1. This taxonomy is defined and compared with the models

found in the current literature, as show the Section 2.2. In Section 3, the criteria and definitions of the generic model of privacy control proposed for pervasive and ubiquitous environments are analyzed in depth. Section 4 presents a scenario outlined of the application based on the characteristics and definitions listed in the taxonomy and model of privacy. In Section 5, a test-bed was employed to validate the architecture of the proposed application for pervasive and ubiquitous environments. Finally, the conclusions and suggestions for future work are described in Section 6.

2. State of the Art

In pervasive environments, there are several problems and challenges that have to be faced, among which, the control and management of privacy stand out. There are several different concepts and definitions of privacy in Ref. [6]. We can cite the ideas introduced by Refs. [7, 8], where privacy is considered to be an abstract and subjective concept that is closely bound up with each individual's perception of what it represents. According to Ref. [6], privacy can be related to providing protection from threats to one's personal property, or physical and moral integrity; these needs are not uniform and are influenced by cultural factors such as religion, tradition, customs, education, and the political environment, as well as more personal factors like age, health, occupation and humor, among others.

Despite the extensive literature in the area, many questions are still left unanswered, while others still require a great effort to integrate several concepts and techniques into a single. A solution can handle privacy in complex environments. It can be readily appreciated that it is not possible to address every aspect of so many situations, and that this invalidates any definition of a specific privacy context [7].

A context is characterized by data that overlaps both the physical and virtual worlds. People do not usually regard physical environments (the office, shop floor or stadium) and virtual environments (the computer desktop or mobile phone menu) as separate entities, since objects and processes can be represented in both worlds. Hence, it is necessary to project structures that are capable of representing elements from both the real and the virtual domain. These elements should be represented in a way that is as generic as possible, to assist the creation of environments that can provide better support for associated physical and virtual tasks. This can only be achieved by putting forward taxonomic definitions that allow the isolation of specific parameters and requirements associated with pervasive privacy, a task that will be discussed in the next sub-section.

2.1 Taxonomy of Privacy in Pervasive Environments

In this section, we outline a taxonomy for privacy in pervasive environments, as shown in Fig. 1. This is based on the current literature and extends the concepts of privacy by taking into account the context of the environment.

In Ref. [9], a few important requirements were set out, in which the pervasive user is described as:

- Collaborative: The user should provide access to information and services (such as music, videos, personal data, and location) in a collaborative fashion, in order to enhance both his own experience and that of other users, as well as to improve the system in general.
- Flexible: Users can adjust the level of collaboration to suit the safety levels required by a certain service request.

Thus, as a result, there is flexibility between the users and sources of information.

- Visible: The user provides access to his profile and identity, which may hold several classifications as described in Ref. [9], where the user's identity may be weak (with a minimum degree of trust), average (medium level of trust) and strong (high level of trust).

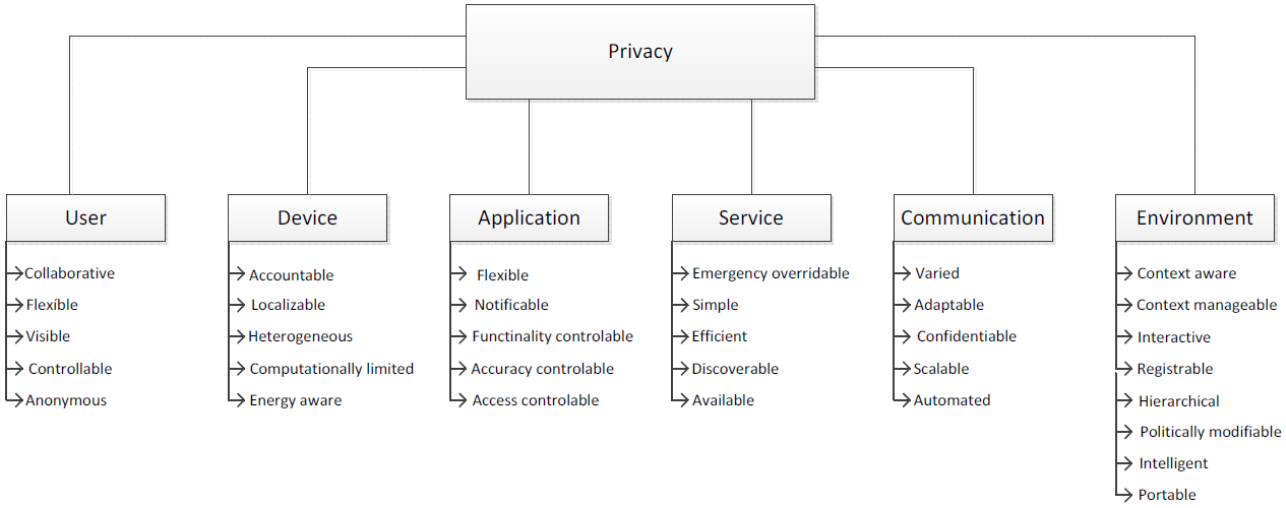

Fig. 1 Taxonomy of privacy in ubiquitous environments.

Other characteristics should also be considered, including anonymity; there are several situations where the user's location cannot or should not be divulged, owing to the nature of his/her occupation or for personal reasons.

• Controllable and the sharing of data should be controlled by the user. His opinions, characteristics and personal data may change at any time, depending on his everyday decisions and lifestyle.

• Anonymous: Even if user management modules have access to many types of information about users, the user should always be in a position to decide if he/she does not wish to provide access to certain resources or services to other users anymore, by changing his profile or context.

The work carried out by Ref. [10] shows a control mechanism applied to a music sharing service based on the location of Wi-Fi hotspots. This paper adopts a different approach from other privacy studies, since it envisages the possibility of defining types and sizes that can be transferred, depending on the current Wi-Fi spot and location. However, this approach does not handle the environment itself but is only concerned with the Wi-Fi spots within it.

In Ref. [11], there is a different solution based on an algorithm that computes an area that depends on the level of data protection required. The use-case was a hospital environment where user data and location could be shared.

However, the study did not predict interactions with the pervasive environment, but only took account of its location [12]. The study also adopted different approaches that could restrict access to certain information, such as document validation.

With regard to desirable characteristics for handling privacy in devices, some works investigated user locations by means of GPS, wireless access points or cellular antennas, and made use of coordinates between locations to control access to services within the pervasive environment. The research conducted in Ref. [13] defines the following taxonomic requirements for devices:

• Accountable: These devices must be registered in the places they frequent regularly, since this helps to reduce the amount of unnecessary data traffic and helps speed up the device identification system. An actual application of this would be the storage of device characteristics that visit a certain environment, such as login credentials or available services.

• Localizable: Direct access to a data base containing information about users and their devices is necessary to validate basic information. Treatment of other types of information should be handled through the access point that the user is currently

using, in order to ensure security and reliability.

• Heterogeneous: A single device may use different kinds of communication protocols and provide different kinds of services.

• Computationally limited: This is a very important factor when dealing with battery-powered or mobile hardware, and it should be taken into account when designing effective privacy mechanisms [14]. Low power processing or limited storage should not be regarded as limiting factors but as challenges to be overcome to achieve computational balance.

• Energy aware: Depending on the hardware in question, we need to consider the energy consumption used by the application layer, service layer and communication. A pervasive system should always take into account how applications or mechanisms can help reduce the amount of energy consumption. The architecture of the system should always focus on external information processing, leaving the sensing and transmission tasks to more limited devices.

The taxonomy definitions for applications and services are based on the work described in Ref. [7], which outlined the desirable requirements for privacy services:

• Flexible: Users should be capable of defining their own privacy preferences, with different levels of detail for different groups of people. Different kinds of users may have different needs while different groups of people may wish to share information in distinct ways (groups united by their religion do not share the same kind of information as groups united by an interest in sports).

• Notifiable: The users may want to be notified of, or to scan, any attempts to gain unauthorized access to their contextual information. Hence, it should be possible for the user to create custom notifications for different situations. While in certain scenarios, it may be useful to be notified of every access attempt, other scenarios may require these warnings to be automatically discarded (for example during sleep times, which are not the same for every user and may vary every day).

• Controllable functionality: In addition to the access control options (“grant” and “deny”), a third option (“not available”) should also be made available. This option allows users to deny access without the requester being aware of it. This technique is also known as “plausible deniability”.

• Controllable accuracy: Users may adjust the temporal and spatial precision of their context information. This usually applies to the user’s mobility, availability and daily tasks, where information is constantly being changed or updated.

• Controllable access: The users should be able to block access to any contextual information at any time. As a basic security routine, the system itself should be able to issue a warning that the user or his device are under a security threat and block the sharing of any kind of information. The user may also find himself in unknown public places where information sharing is not recommended.

According to Ref. [15], traditional autonomous computing and small networks depend on user authentication and access control to guarantee security. These interaction-dependent methods have certain rules and regulations that restrict the ability to access, use, modify or visualize resources. However, mobile users need to be able to access hosted resources and services at any time from any place, which leads to serious security risks and access control problems.

With these challenges in mind, Ref. [12] proposes a solution based on the management of trust which involves the adoption of security policies and the assignment of credentials to external entities. These credentials are checked to see if they conform to the defined policy, while the level of trust in each of these entities is validated through third-party input (the feedback from other entities, for example). The level of trust attributed to an entity can be correlated with the level of access they are given. Despite being valid, the proposed solution does not handle all the necessary requirements shown in Fig. 1 of its

taxonomy, or does it focus on the pervasive environment.

Some other characteristics associated with pervasive Services and applications are shown in Ref. [16]:

- Emergency overridable: The users should be able to define the exceptional policies that precede any other kind of privacy policy. Just as in the real world where we face factors that are beyond our own control, it is necessary to define rules that must be complied with in priority situations, such as emergency phone calls between family members at inopportune times or from call-restricted locations.
- Simple: Another point is that the users should not be bothered with too many configuration options for their privacy preferences. Basic usability guidelines suggest that no one wants to navigate through a lot of interfaces and menus to configure a particular functionality. Thus, the system should be able to store relevant configuration information, for example the user's most accessed functionalities, which can then be used to generate useful "shortcuts".
- Efficient: The handling of privacy concerns should not cause a significant delay in communication or a heavy processing load for the context providing services;
- Discoverable: The application should meet the necessary requirements and provide the parameters for the discovery and offer of available services to the users. The discovery of services and the mechanism to make them available should be both omnipresent and automatic, in the sense that it should not be necessary to reconfigure the parameters of a device for each new situation.
- Available: The application should have control over the usage of services, to ensure that all users, devices, communications and services in the pervasive environment can enjoy equal access to a greater amount of information.

The work carried out by Ref. [16] shows that several problems were encountered when dealing with the management of security in pervasive applications and services. However, we found that its taxonomic description of communication, based on protocols and services, represents its most distinguishing feature. The taxonomy to communication is described as:

- Varied: The communication should adapt to as many distinct devices and environments as possible, and should be able to exchange data by means of different communication media without the need for user intervention.
- Adaptable: The communication should control which protocols can be used in each environment, to reduce the risk of message losses and processing requirements.
- Confidentiality: The infrastructure must be capable of handling performance, certification control, login mechanisms and other management functions so that it can provide pervasive communication in a secure and reliable way, as described in Ref. [17].
- Scalable: A communication protocol should simultaneously serve as many users and devices as possible and provide services in many different environments while maintaining a satisfactory level of quality.
- Automated: The communication should be able to support mobility and remain adaptive by using unicast, broadcast or multicast communication channels, without the need for user intervention.

Even in the area of taxonomic descriptions for pervasive computation, the work in Ref. [18] establishes a framework and middleware architecture for pervasive computation. This study argues that a fundamental demand for pervasive computing requires the automatic physical integration of hardware devices. However, this work treats the infrastructural requirements of pervasive software in a general sense, and does not specifically address the question of pervasive environments from a computational standpoint.

The work in Ref. [19] explores different forms of communication and distinct infrastructures that

support several requirements and characteristics for pervasive computation: scalability, heterogeneous environments, integration, contextual invisibility, awareness and contextual management, which have been described as the main challenges that had to be addressed by pervasive computation. The work described in Ref. [20] addresses the modeling of systems in the area of pervasive computation. It contains a study on the issue of privacy, which is used as a means to extend previous research work that devised a meta-model to be used as a basis for the construction of ubiquitous systems. The extension proposed in Ref. [20] seeks to specify privacy at user-level for ubiquitous environments. Although, it can be claimed that this study has made a considerable contribution to the state-of-the-art, the proposed approach does not directly deal with privacy in the ubiquitous environment, but rather, is concerned with the ubiquitous user within the environment. Moreover, we lay down a few basic requirements for pervasive environments, which include the following:

• Context aware: It is one of the most important factors in intelligent environments since it makes a ubiquitous system as minimally invasive as possible. The system and the environment should be able to recognize the user's current status and adapt their behavior accordingly, as described in Ref. [2]. For example, a user that enters a pervasive space should be automatically identified and have access to the services and environmental configurations that should be available to him.

An interesting case of the importance of privacy in pervasive environments is described in Ref. [21], where a British woman found out about her husband's infidelity through Google Street View, thanks to the customized number plate on the husband's car. Since its launch, Google Street View has been the target of complaints and was severely criticized for (accidentally) obtaining pictures of people performing acts meant to remain private, without their consent or knowledge. If one thinks of the world of pervasive computation and sees the car as a pervasive environment, it is possible to configure the car in a way that prevents its location from being published. This could be carried out, for example, by equipping the car with a RFID chip that is read by the Google Street View's vehicle, and then either gives or denies permission to photograph it. This 2009 case proves that privacy is neither a novel nor a trivial issue.

In an attempt to establish a reference model, Ref. [22] provided a taxonomy that was aimed at establishing a new set of QoS metrics for classifying and characterizing WSNs. However, the work did not consider privacy or metrics control. The work in Ref. [23] carries out a review of the state-of-the-art in privacy preservation techniques and a taxonomic analysis of the control of privacy and contextual data for WSNs. Two main categories of privacy-preservation techniques are discussed—data-oriented and context-oriented. This work is particularly useful, since it solves the problem of ensuring privacy for both data in the network and the application context, but, on the other hand, it does not address the challenges associated with the application environment or the pervasive system as a whole, since it is wholly focused on WSNs. We can draw up a new privacy agreement on the basis of this work:

• Manageable context: The environment should allow the user to share his own data and use the services and information made available by other users and the environment itself. Thus, support for a domain-independent representation of services and information is expected from the environment. With help from the pervasive environment, the user can choose the most suitable and context-appropriate services to achieve his goals. In view of the heterogeneity of possible devices and configurations in ubiquitous computing, the provided services should be accessible from anywhere within the environment and available in as many different formats as possible.

• According to Ref. [20], an understanding of the

nature and context of human activities is a very important research field in many areas such as psychology, sociology and ergonomics. However, this wide range of involved areas can cause conflicts, since each area offers a different perspective and can propose and explore a different strategy for a better understanding of human activities. An overall understanding of the subject can only be obtained by conducting research in each of these areas; this overview is very important to enable ubiquitous computation to accurately detect, represent and analyze human activity, which is a multidisciplinary challenge of considerable complexity. This reveals the need for combining different systems as a means of providing the ubiquitous environment with information and guidelines on how to act based on human responses. As a result, we can identify a new requirement for obtaining a privacy solution:

(1) Interactive: Users must interact with the environment in order to obtain information about it. This interaction should be intuitive, pleasant and adjusted to the environment context. Pervasive computation can lead to a good deal of inconvenience, such as intrusive advertising mechanisms: Most people do not wish to keep being informed about products for sale whenever they pass by a store. One way to circumvent this problem is to only inform registered users that have subscribed to certain types of notifications that match their hobbies and interests, and to allow them to disable and re-enable these notifications at any time.

(2) Activity recognition: The user activities can be effectively recognized through specialized activity recognition mechanisms. This information can be used to improve that ability to infer the user's context from the pervasive environment. Context inference can be used in numerous situations, such as employing automated network mechanisms to suit the user's needs (e.g. choosing the best wireless interface available, depending on the user's location and activities).

(3) Registrable: Privacy management in a ubiquitous environment should allow technology to remain very close to individuals and operate in a variety of real scenarios. For the control and registration of environments, it is necessary to have information that describes everything that belongs to the environment, as well as accessibility conditions, available services, shared resources, authorized individuals, devices, communications and applications that allow interaction to occur. However, the rules that govern privacy and control access should always be based on the environment and its definitions.

According to the literature, the ubiquity paradigm compels the computation to be invisible, that is, to carry out its operations with the minimum distraction from the task in hand [1-3, 7, 20, 24]. The environment should not have to auto-reconfigure at each login solicitation or change of users. In other words, the environment should be configured in a way that is unique and inherent to its nature and context. If a user goes to a soccer match and sits on the opposite team's bench, this represents a serious security event for the stadium's pervasive system, since the user's location might be hazardous to his well-being. Thus, we need to define the environmental hierarchy of usage:

- Hierarchical: Pervasive environments should be governed by a set of established rules, much like those in the real world. In general, each environment is managed by a high-ranking entity, which might be the environment's owner or the person in charge. Companies, churches, schools and even our own homes are governed by a hierarchical tree that determines how individual elements relate to each other in terms of authority. The pervasive environment's hierarchical taxonomy should follow the same structure as the hierarchies that we find in real world locations.
- Politically modifiable: With regard to the policies of the environment, it is necessary to enable the user in his current location to have control of input/output

of information, that is, the user's environment controls the way information and services are shared. For example, a classroom can be configured to receive the teacher as a kind of advanced user and students as simple guests [25].

In the research conducted by Ref. [26], a logical language for expressing security policies called LEPS is set out, which defines a security policies model for access control services, authentication, integrity, privacy, auditing and non-repudiation. The proposal is of value, but it remains focused on users and related groups. However, we can use the concepts introduced by LEPS to create other requirements from the hierarchical model proposed:

• Intelligent: These mechanisms should have definitions and rules so that the environment itself is enabled to make the decisions intelligently, without human intervention. One of the most popular solutions is the introduction of artificial intelligence mechanisms such as those outlined by Ref. [27], where an inferential information system called MANFIS is described, which allows multiple data input and consequently only one output. In research conducted by Ref. [14], there is a classification taxonomy for different ubiquitous environments, which is supported by two main categories: interactive environments and intelligent environments. The taxonomy classifies all the types of ubiquitous environments which allow interaction with the user in intelligent daily operations. The classification is based on the routine behavior of the user, so the environment has the usual information. The study states that, regardless of the environment, decision-making mechanisms are necessary to maintain control [28]. However, there is a lack of a precise taxonomic definition of what is required of pervasive environments. The authors simply draw on the studies and techniques employed by other researchers to give an idea of what would be an ideal solution. They fail to address the various issues of pervasive environments, and simply conclude that the pervasive environment should be iterative so that it can meet the pervasive requirements.

• Portable: In the case of the pervasive/ubiquitous environment, portability is an important requirement that was not addressed by any of the investigated works. It is necessary because restricting an application or service to a single programming language, operating system or other forms of use in pervasive environment, can also be considered to be a kind of sharing and is not an imposition.

It is impossible to have pervasive/ubiquitous environments that impose the use of certain technologies, devices or software. Certainly, there are situations in which some regulations are necessary when dealing with the topics discussed above, but in these cases it is necessary to find generic solutions where the user employs as few of the environment resources as possible.

The work in Ref. [29] proposes a model, based on theoretical investigations into personal interrelations, that seeks to embrace human privacy and bring it to the pervasive world. A state of the art that is enhanced by works on human interactions, is used to derive the peoples' preferences from data on groups of users and the collaborations between them. The model proposed by Ref. [29] is based on user registration and control, where the environment acts in an omnipresent way, and also on the stored information. While the idea is well founded and has several factors that are of value to this work, the model does not address the question of the pervasive environment, but focuses on the people who inhabit it, and is shaped according to their preferences.

In Ref. [30], research is conducted into several privacy issues addressed in the context of HCI (human computer interaction) and, based on this research, a number of trends in the area are defined. As its main contribution to this research study, the work addresses several questions, including the protection of the pervasive environment. We examine this item (and how it can be adapted) from the perspective of privacy

control in the environment, since the focus of Ref. [30] is not on the privacy of pervasive environments. The work in Ref. [31] offers a solution based on authentication that takes account of various technological scenarios, such as RFID use, while suggesting a single mechanism to manage different authentication protocols in ubiquitous environments. However, these mechanisms are only concerned with performing authentication iterations over the pervasive system, and taking note of possible changes in the environment.

These limitations make it necessary to add additional systems so that the pervasive environment can share information, and this restricts the feasibility of adopting the proposal. There are some other examples that have a bearing on the aims of this study with regard to control of privacy and, hence, the sharing of data, describing the privacy of obtained information and the classifying the captured flows as public or private. Some early work had already expressed concern over what is a recurring problem [2].

2.2 Comparison Table of the State of the Art

Among the studies reviewed, there is a focus on devices as well as their means of communication, and also on the relationship that is necessary for control of privacy. However, these studies do not deal directly with the privacy control of the requirements and relationships within the environment. In view of this, we seek to define two key areas, the handling of environmental characteristics and the sharing of information that will be made available to the user. Devices and users will have to adapt to the environment context and not the other way around. This approach aims to adapt the pervasive context to the real world in which we live in, where a certain environment and its rules are not changed due to the presence of the users within it and their privacy preferences. On the basis of these premises, some works were analyzed, and prominence was given to those that [12] calculate the user's location by means of the GPS and computes the possible points inside a building, where a particular user may be. It then applies privacy rules to the users, depending on their probable location [12]. While it was an interesting approach at the time, there still remain several drawbacks to this approach. It does not address, for example, the question of the services carried out by the user since it does not know his exact position, nor does it handle the data sharing in larger areas, since the GPS may show some discrepancy between the detected and actual position, in the order of several centimeters. From an application standpoint, this may seem a minor point, but in the case of distinguishing between different divisions, a thin wall that is a few centimeters thick can make the difference between one environment and another.

In Ref. [32], service discovery protocols are designed to reduce administrative overhead and increase usability. They can also save pervasive system designers from having to predict and encode all the possible interactions and the states between the devices and applications at design time. By adding a control layer, service discovery protocols seek to simplify the system performance. This work shows good taxonomic definitions of communications and services. However, the proposed solution is focused on control protocols and data generated by the device connected to the user, and does not provide definitions and descriptions aimed at controlling the pervasive environment. A comparison of different research studies is shown in Table 1 to achieve a better overview of the state of the art and the proposed model. The Table shows several of the requirements for pervasive/ubiquitous environments collected from the researched literature. The research explores these requirements extensively and provides taxonomic definitions for each, as shown in Fig. 1. On the basis of this study and the state of the art, we have defined a model for privacy in pervasive/ubiquitous environments, called the GMP (generic model of privacy)

Table 1 Comparison between work-related.

Approaches	User	Device	Application	Services	Communication	Environment
LPPC	Approached	Not approached	Approached	Not approached	Not approached	Not approached
SDPCE	Approached	Not approached	Approached	Approached	Development	Not approached
GPASCRM	Approached	Not approached	Approached	Approached	Approached	Not approached
TBSPCE	Development	Approached	Approached	Approached	Not approached	Not described
EPAECU	Development	Not approached	Approached	Approached	Approached	Not described
TGSIUC	Development	Development	Approached	Approached	Development	Not approached
LEPS	Approached	Development	Approached	Approached	Approached	Development
SLPPC	Approached	Approached	Approached	Not approached	Development	Not approached
INFOPOINT	Approached	Approached	Development	Approached	Approached	Not approached
PRISM	Not described	Not approached	Development	Development	Approached	Not approached
MGSPPSC	Approached	Approached	Approached	Approached	Not approached	Development
TSPPC	Not approached	Approached	Approached	Approached	Not approached	Not approached
ECMPPCE	Approached	Not approached	Approached	Development	Approached	Not approached
ASCLP	Not approached	Not described	Development	Development	Approached	Not approached
FSSD	Development	Not approached	Approached	Approached	Approached	Not approached
SDPCE	Development	Not approached	Approached	Approached	Approached	Not approached
TUCE	Approached	Approached	Development	Approached	Approached	Not described
CAUASPB	Approached	Development	Approached	Approached	Approached	Not approached
EUPHCI	Approached	Approached	Approached	Not approached	Approached	Not approached
GMP	Approached	Approached	Approached	Approached	Approached	Approached

(Section 3.3), which is compared with the other models shown in the Table 1. We have used a number of definitions, such as:

- Approached: The work deals with the item;
- Not Approached: The work does not address the item;
- Not Described: Information on the item is not found;
- Development: The item is still under development; this often pointed out in the testing, validation, results or future work sections.

Several works describe a particular solution that may be applicable to the pervasive environment, but fail to provide information, testing results or simulations on how to control the pervasive system and all its elements that are found in the environment. In Section 3, we will outline a privacy control model for pervasive/ubiquitous environments.

3. Proposed Model

Context awareness, according to Ref. [33], refers to any information that can be used to characterize the situation of an entity. An entity can be a person, object or place that is considered relevant to the interaction between a user and an application. Also, there are four types of context that are defined by Ref. [8]:

(1) Context of Computing (networks and resources);

(2) User Context (people, places and objects);

(3) Physical Context (lighting, odor, temperature);

(4) Temporal Context (hours, days, months).

An example of use context is the ability of a device to measure the temperature in a given environment and employ equipment (e.g. air-conditioning) to provide the ideal temperature for the users inside. Another definition is given in Ref. [2], which states that the context of a user in context-aware applications consists of attributes such as physical location, the physiological state, the emotional state, personal history, and daily patterns of behavior, among others, which, if applied to a human assistant, can be used for decision-making without the constant need for the user's attention. However, there are two serious difficulties related to the development and use of

context-aware applications: the complexity of providing context-aware services and the need to maintain the privacy of contextual information (e.g. the location of the user).

These applications generally involve the use of computational contexts (e.g. energy level, bandwidth), personal (e.g., profile, user location) or physical contexts (e.g. temperature, humidity) to provide the customized services that are most appropriate for a particular end-user [7]. In Europe, there are already laws and policies designed to protect the privacy of personal data. For example, the global roaming service for mobile phones encouraged some countries to implement legislative policies aimed at protecting personal privacy. The European Union Directive on data protection [34], which currently comprises the most complete set of privacy laws, has had several updates since the work described in Ref. [7] was published. These updates state that personal information should be:

(1) Obtained accurately and within the conditions stipulated by law;

(2) Only used for the original purpose specified;

(3) Requested in an appropriate manner, that is relevant to the original purpose; i.e. the accuracy of the requested information must not be more specific than what is absolutely necessary to meet the needs of the requester;

(4) Kept in a safe place;

(5) Accessible to the owner of the information;

(6) Destroyed after the purpose of its use has been fulfilled. Some other new policies have been added since then, with an emphasis on public protection. These establish a mandatory legal framework that guarantees the individual right to privacy [34]. This right is ensured through the implementation of measures that must be respected by any organization (including governments and corporations) that deals with personal data during the stages of both the application's design and its implementation.

These measures cover the processing of personal data and include provisions relating to the following:

(1) security of networks and services;

(2) confidentiality of communications;

(3) access to stored data;

(4) processing of traffic data with location and identification;

(5) personal control of subscriptions to public lists and unsolicited commercial communications.

The essential criterion that allows data to be stored and processed by an organization is an effective agreement by the individual when providing his data. The policy covers all data sent over public networks in Europe and, therefore, also covers the data or services that originate outside Europe. In the light of these considerations, this paper provides a model for user control in pervasive environments formed by the adjustment of profile settings, where the pervasive environment context profile must be adjusted to control the privacy of users based on the characteristics of their lives and on rules laid down by Ref. [34].

3.1 Criteria for a Model

A model for the pervasive environment context was devised for this purpose and will be outlined below. In a ubiquitous environment, it is necessary to draw up criteria for user authentication. For this reason, we have defined the following controls:

(1) Blocked: Access should be blocked to users for fixed or indefinite periods.

(2) Guest: Access should be limited and controlled;

Access can be made available for a fixed or indefinite period.

Restrictions on services and sharing; Controlled privacy availability.

(3) Basic: Controlled access.

Sharing of resources and services is limited by the environment;

The limiting factor will fall within a scale ranging from 1 to *N*, depending on the environment and its resources, his requires access to an ubiquitous data

base where all available resources, in all pervasive environments, are registered;

Sharing of location between other users of the pervasive environment.

(4) Advanced: Access to all previous levels;

Complete access to all resources and services in the environment.

(5) Administrative: Access to all previous levels;

Full control and management of the pervasive environment.

These criteria will be assigned by the ubiquitous environment to the user that requests authentication in it. However, when dealing with external pervasive environments such as parks, squares or other public places, the user receives administrative access level, since no ubiquitous environment should exert dominance over a pervasive public environment. On the basis of these criteria, we conclude that it is necessary to adopt a middleware for the control and management of different environments and configurations. In Subsection 3.2, we describe the proposed architecture that will be used for this kind of middleware.

3.2 Middleware Architecture Model

The (MW) overall architecture of the middleware supports all the necessary levels of control of the application, software and hardware, and will be based on the initial model proposed by Ref. [35]. In this model, there is a middleware focused on pervasive systems and divided into four layers: hardware, software, middleware and application. In this architecture, developed modules required for initial tests of a pervasive environment were validated using an OMAP platform [36].

Based on the above references, a few models were implemented in the tests that were carried out, namely a context management module, using a pervasive scheduling system as an application scenario, where users accessed their schedule in an intelligent and ubiquitous way. However, this architecture was not designed with applications involving WSNs or RFID. This observation can be confirmed at the time when protocols were implemented for pervasive environments with the aid of RFID and WSNs in the work conducted by Ref. [37].

In this application, it was possible to reduce energy consumption in wireless sensor networks with the widespread use of RFID tags in some parts of the pervasive environment and through a solution based on the ZigBee protocol. After identifying these deficiencies in the MW proposed in Ref. [35], we noticed the need for changes in the structure of middleware to give support for pervasive/ubiquitous applications and to manage RFID and WSN protocols and devices. Owing to these failings, the authors in Ref. [38] set out a new proposal for a MW platform capable of supporting the necessary technologies for pervasive and ubiquitous environments, with a focus on protocol management and reducing energy consumption, in accordance with the model shown in Fig. 2.

The middleware proposed in Ref. [38] consists of four interconnected layers comprising the characteristics and requirements for the control of wireless sensor networks and RFID in pervasive/ubiquitous environments. One of its main features is the lowering of energy consumption caused by the reduction in the exchange of messages between the nodes and the base station. Thus, the middleware can be used to deal with hybrid problems involving pervasive/ubiquitous environments. The main idea is to allow individual devices to meet the needs of their users or the environment as a whole, by adapting to each environment and its underlying infrastructure to the best of their capability. The following is a description of each layer, as well as their characteristics.

(1) Hardware Layer (HW): There are modules employed for handling the physical requirements necessary to deal with the physical devices and these are implemented in HW, as described:

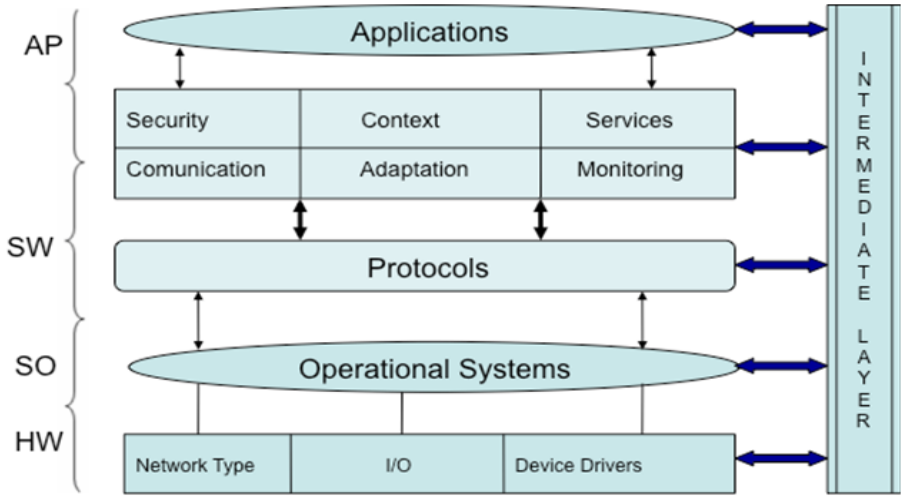

Fig. 2 Hybrid middleware.

• Network type: used to connect the middleware platform to the pervasive network. It also has specific features that will be used to assist and control the functionality of the device.

• I/O: used for communication and interfacing with users, environments and devices.

• Device drivers: pre-processing unit responsible for managing, storing and executing the minimum requirements for the operation of the devices. Acts as a trigger for the connection of physical devices registered in the system, such as MAC address specifications, IMEI (international mobile equipment identity), and Bluetooth, among others.

(2) OS (operating systems) Layer: aims to handle the functions of operating systems for embedded systems. It is divided into two sub-layers: the device drivers sublayer manages the components of the physical layer and the embedded operating system sublayer manages the tasks of the application that runs on the device and determines the services provided by it, while also coping with the limitations of the system. Thus, every change that occurs at the operating system level in the user devices is dealt directly at this layer.

(3) Protocols: responsible for carrying out the handling and management of the data protocols used in pervasive/ubiquitous environments along with the wireless sensor network and other devices such as RFID.

(4) MW (middleware) layer: is a set of components that assist in the integration and treatment of devices by the pervasive/ubiquitous network and carries out necessary services and makes other features that comprise the middleware architecture available. This layer consists of six primary modules:

• Communication: Integrates the device in the network and manages the communication of the device with other devices.

• Services: Carries out the management of services, environmental resources and devices for pervasive application. Another attribute is that it provides and controls the adaptation of new SW components.

• Adaptation: Responsible for the adaptation and management of users, services, devices, applications, communications and pervasive/ubiquitous environments.

• Security and privacy control: Responsible for handling the security environment, providing control and authentication services. The main purpose of this module is to manage and control the various types of sharing, by relating them to the privacy of the environment.

• Context: Helps to detect the context of the user and the environment.

• Monitoring: Provides environmental and device monitoring for the application, by reporting status, errors and problems.

(5) AP (application) layer: is a module that has fragments of the applications running in the environment. This layer is responsible for performing all the necessary settings for any application to run in the environment. An example might be the provision of services made available by a coffeemaker in the pervasive environment where a particular user is located.

(6) Intermediate layer: This layer is intended to connect and interact with all layers simultaneously, and its main objective is to establish a connection between two layers without necessarily connecting to others. For example, a sensor's only purpose may be to alert the application to the occurrence of an event. In such a simple situation, the sensor may have its own self-managing operating system, and does not need to be connected with the operating system and other layers, and thus save resources on the platform.

The treatment of privacy in the environment will be included in the middleware context module, since the context can be handled at the user, device, communication, services and environmental levels. Thus, it is not necessary to restructure the existing architecture, or allow the existing middleware proposed in Ref. [38] to be used again. Basically, there will be a specification within the module that will be called "triggers". These triggers are inserted options that are used to activate the context module. They include the following types of options: environment, user, devices, communication, services, application and others.

This enables the use of the middleware architecture in the generic privacy model that will be described in Subsection 3.3.

3.3 Generic Privacy Model

The use of a mechanism for managing of privacy in environments pervasive and ubiquitous must meet the application requirements. In some scenarios, it is necessary to collect information from users to the system operation. These information should be treated legally and ethically because of the privacy of individuals. We propose a generic model for managing privacy in environments pervasive and ubiquitous as shown in Fig. 3. The proposed generic model contains several components for controlling the environments pervasive and ubiquitous, as described follows:

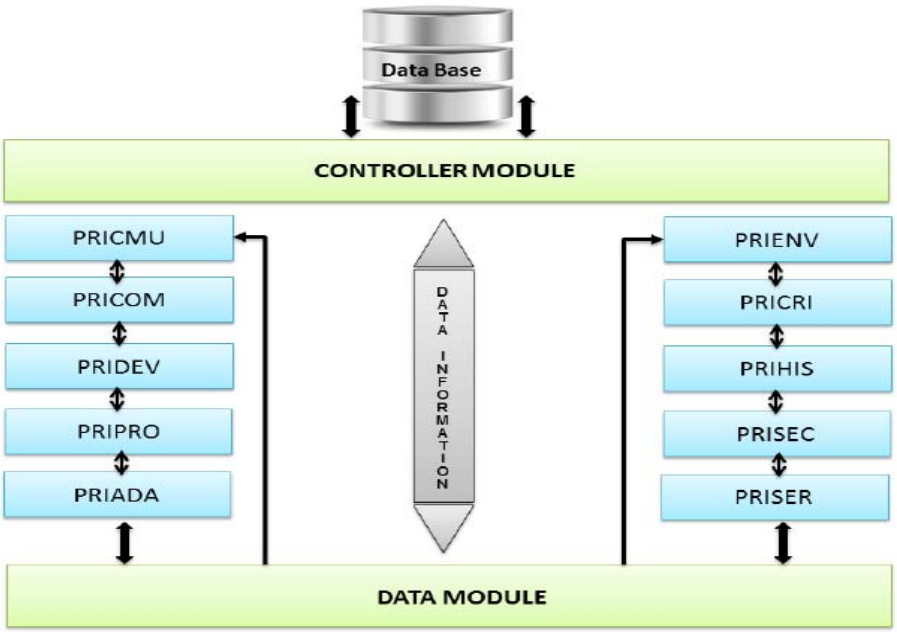

Fig. 3 Generic model for privacy.

(1) Data base: This data base stores information about rules and definitions of user, devices and communications in environments pervasive and ubiquitous. This data base is like a single register containing all information necessary for the control and management of the privacy mechanism. This data base is linked to the data module, represented by double-headed arrows, where information is exchanged with the data module.

(2) Controller module: This module receives the access requests and makes the control of the data base tables directly, according to the requirements and definitions of access and control of environment. It also performs requests for validations and updates the data base after the information has been returned to the module calculated and refined previously by the control module. These refinements are based on the characteristics and definitions in each management module. The data updated and set are returned to the requesting with his permission in accordance with the received variables.

(3) Data module: This module performs all calculations of variables and parameters received from other modules, and generates a single output information for each processing run.

(4) PRICRI: This module defines the rules and criteria of access, use, sharing, location, etc.. These rules can be added, changed, modified and/or replaced in accordance with the environment and with established rules. These definitions are handled individually by other modules that have individual characteristics and controls. The operation settings are preset for each environment and can have variations, such as the same user access the same room on the same day with different criteria defined according to the time of access.

(5) PRIDEV: The module management and privacy control of device has as goal treat the data that is transferred by devices once that such devices may be of the environment itself and of other itinerant device. The management and control are related to the characteristics of software and hardware of each device (size, weight, screen resolution, operating system, media, etc.).

(6) PRICOM: This management control and communications privacy. This module defines the various forms of communication within the environments pervasive and ubiquitous, such as restrictions sign, type of adapter used to the controller accesses like in the environment of the real world, in which certain environments only have one type of communication.

(7) PRIADA: This management control and adaptation module, which is responsible for processing information related to the adaptation of software and hardware in environment pervasive and ubiquitous. For example, treatment of content and media to be used in different devices by presenting differences in performance, functionality, communication or configuration.

(8) PRISER: These modules service management environment. This module treats of the information about the availability of services to be used individually for each environment. For example, the information shared with other environments, such as the devices, communication types, location of users, environment features and components that interact with users. The definitions and rules for the use and availability of these services are inserted into the module environmental criteria in order to control access and management.

(8) PRIHIS: This module stores and treats the information about the user history, environment, devices and other variables that may be added later with the goal of obtaining contextual in formation. The main feature is the use of information captured over a given period based on other sources of information, as for example, multiple tracks, context, etc..

(9) PRIPRO: This module is performed the transaction control on the management of user profile. Its main objective is control the information, previously defined by a search engine, which has only

the purpose of distributing and direct the synthesized information to the next modules, in order to adapt appropriately to individual privacy based in the individual profile.

(10) PRISEC: This module makes the control and management related to security, both the user as the environment. The module receives the parameters and related data encryption or other security-related settings treatments and forwards them to the requester according to the need for each situation. For example, when entering in a given environment the user may be blocked by situations that are beyond the criteria set within the same environment as the date and time allowed.

(11) PRIENV: This module is registered the attributes related to the environment. Thus, with this information, it is possible check and manage what makes up the environment as well, their capacities and resources in order to share them to users who need according to your availability.

In the next section, it shows an application scenario that takes account the model proposed.

4. Application Scenario

In Ref. [32], the authors developed the Percontrol, a system that automatically manages and keeps track of user attendance. This system detects the entrance and exit of users within an academic or business environment. Percontrol also improves user discovery and localization service, within the local environment based on Bluetooth, Wi-Fi and RFID identifiers. However, the initial versions of Percontrol did not anticipate the use of WSNs or ANNs [39], such versions only intended to automatize student attendance tasks in classrooms.

The application scenario shows the potential pervasive and ubiquitous computation for improving efficiency in workplaces. It also attempts to illustrate different possible perspectives that can have on a single pervasive scenario. This work proposes an extension of the work developed in Refs. [39, 40], increasing the pervasive functionalities available in this user tracking system, with the objective of increasing control over environmental conditions through user's mobile devices. Using SunSpot wireless sensors [18] and Arduino kits [40], Percontrol can sense and manage the temperature and luminosity of an environment; and by using ANNs, the system can also attempt to adjust the values of these environmental properties to fit user preferences and the number of people in the environment, turning it into an intelligent location.

The sequence diagram in Fig. 4 shows the primary inter actions between all parts of the system, as well as the messages exchanged since a user is detected until the environment adapts to its preferences. When the application detects the entrance of a device in the environment, a web service that manages the associations between users and devices is accessed. The device is identified through its BDA (bluetooth device address), Wi-Fi or RFID. The application maintains a module called BlueID which holds a list of all devices that were ever detected. Each time the application verifies the devices currently present in the environment, it performs a comparison with the

previously stored list; newly detected devices generate an "entry' event while missing devices are associated with an "exit" event.

When accessed, the web service returns the username to the application, and also associated device resources and personal preferences through the HTTP protocol and an XML format message. The application also communicates with the SunSpot sensors to fetch the room temperature, luminosity, humidity or other environmental data that may be used at a later time. The following format was used to communicate with the sensor: messageID and sensorType. Both messageID and sensorType are numerical values. The messageID field is used to associate sensor response with the respective BlueID request, an important step since communication is asynchronous.

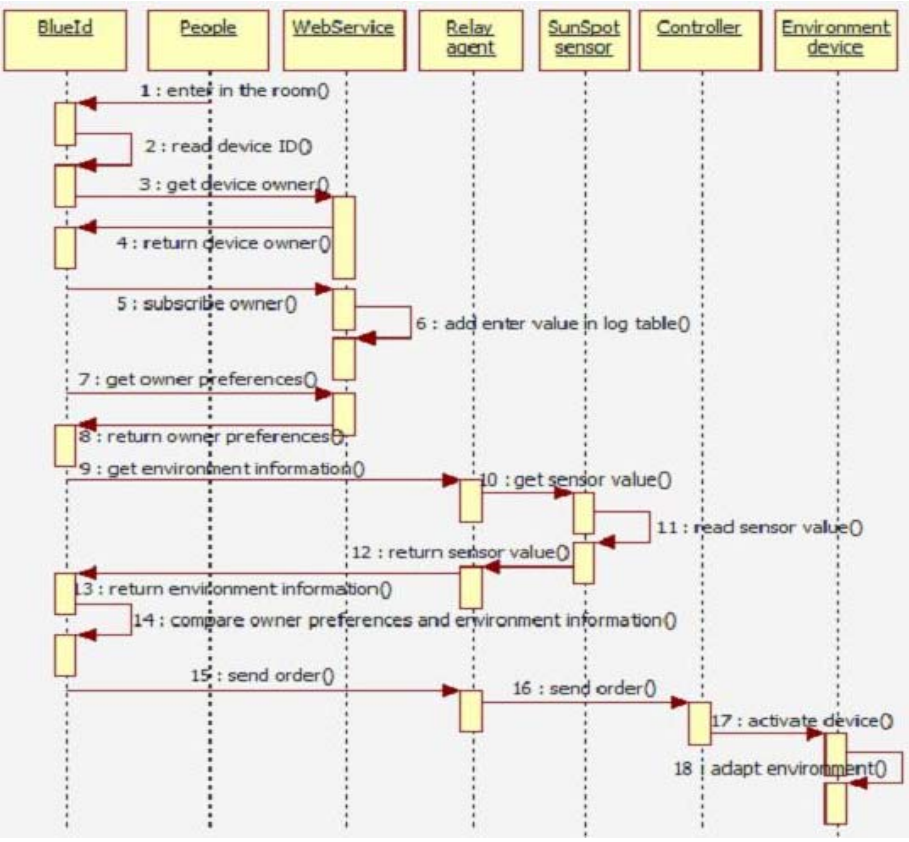

Fig. 4 Sequence diagram.

The sensorType represents the type of data being sent (luminosity, humidity, etc.); the "\n" character is used to mark the end of a message, while the ":" character is used as a data separator. The current environmental state is compared with user's environmental preferences in order to decide the needed changes to be done. After a decision is reached, the environment sends commands to the actuator controllers, connected through USB to an operating computer, to change the environmental characteristics (e.g. turning the A/C unit on and change the room temperature). The extension of Percontrol's functionalities translated into a more complex architecture as shown in Fig. 5.

Initially, the prototype application and its respective transmitters were tested with a Windows operating system, an environment that benefited from the use of SunSpot sensors [39] and Arduino hardware [40]. There were many other advantages that led us to choose the Arduino boards, namely the embedded input/output ports, low cost and strong modularity. The main idea behind the use of Arduinos was to test their viability for middleware development in pervasive environments, not excluding the possibility of having these boards completely replace several individual sensors for an integrated, single board solution connected to a computer. Fig. 6 illustrates the operation of Percontrol.

From Fig. 6, it can be seen that a central controller is missing for allowing the exchange of profiles in the environment, according to user preferences. The main challenge here is controlling the number of parameters generated by the application while using WSNs and ANNs; the number of parameters increase with the amount of nodes and this means larger energetic and resource demands, as well as an more complex neural network processing, which may compromise the ANN's response time.

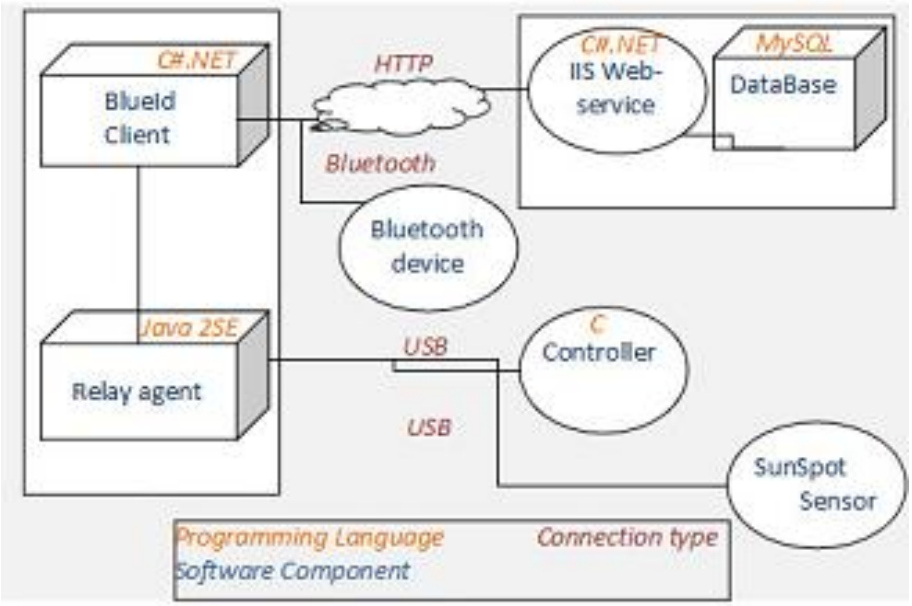

Fig. 5 System's architecture.

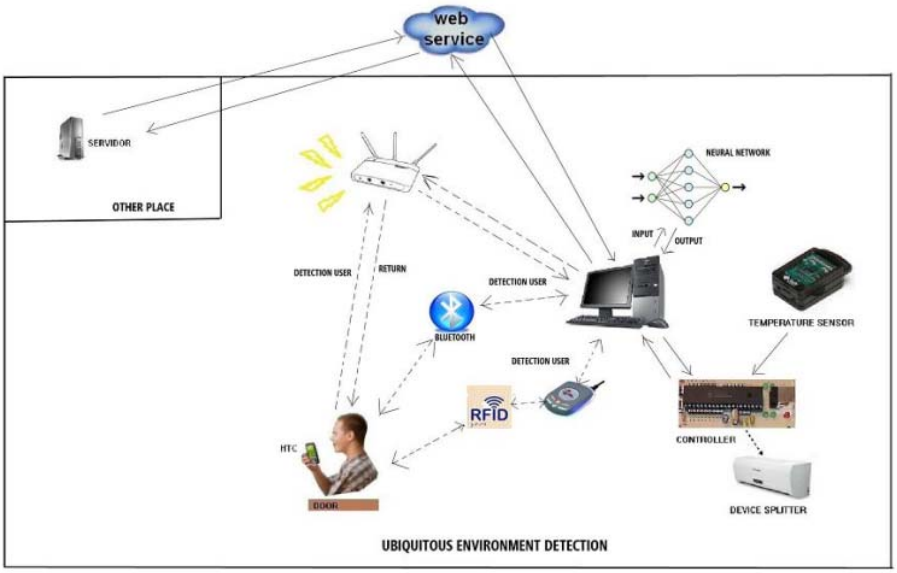

Fig. 6 Functioning of percontrol.

An efficient monitoring of the network performance is necessary to guarantee a good quality of service. An example of this can be observed in the amount of time necessary to obtain information regarding a monitored environment; if it takes too long to obtain environmental information, this information may lose its value from an application perspective. Management of performance may provide means for the application to define proper quality metrics. These may be influenced by node density, exposition time, amount of dissipated energy and other factors.

A mechanism that evaluates the level of importance of information is necessary for the management of quality of service. For example, a sensor detecting a temperature of 20 °C during spring is a normal occurrence, but the measurement of 50 °C under the same circumstances is an abnormal event, which would turn it into a relevant situation that would

require extra attention; it would implicate an artificial intelligence mechanism that could compare the abnormal value with other measured values by other sensors to see if the information is reliable and determine the proper course of action. Information that is of great importance to the normal function of the system should imply a greater effort for proper delivery. That is, energy consumed in communications should vary depending on the importance of the data.

Another relevant management aspect concerns the installation of ad-hoc networks in unknown areas, where the behavior of wireless communications can be highly unpredictable, with high error-rates and considerable delays which may compromise the value of the information provided to the application. Performance management usually includes quality assurance, performance monitoring, control and analysis [41]. The QoS management process begins with the detection of performance degradation and ends when the source of the problem is ceased or removed. In between, the process has many intermediary stages of situation analysis [22]. Initially, there were used only 2 sunspot tests kits containing two wireless sensor nodes communicating with each base station connected to the computer via USB. Thus, it comes the need to conduct a comparative study of routing protocols for use in different environments composed of wireless sensor. To this end, several techniques exist to treat this problem and also allied service discovery, one of the most important, by the SLP (service location protocol) [18], which basically consists of maintaining a directory that contains the services available to whom it is offering them to. However, it is necessary to study thoroughly the operation of routing protocols in order to verify the protocol that best fits the pervasive control system, it is not in the scope of this work—the study of routing protocols. Therefore, for this work there were conducted only some tests to validate the survey and obtained results that demonstrate the feasibility of work and their implementation and use with Percontrol, contributing to the improvement of the system and data so that other researchers can use it.

5. Example of Use and Preliminary Results

One issue when having multiple users on the same system is the problem of concurrent data; e.g. the configuration of an air-conditioning unit may be influenced by every user that registers in this environment, since each user might have its own preferred temperature, and the temperature itself is general throughout the whole environment. In order to bypass this problem, the decision-making process for selecting the best “average” temperature must take into account the individual preferences from all users within the environment. A widely used solution [37, 42, 43], that has shown great results is the use of AI (artificial intelligence), in particular ANNs [44]. A neural network bases itself on real data that has occurred in the past and has been stored within the system for posterior access and use.

The main objective of this work is not the choice of proper protocols or AI tools, but the creation of novel help mechanisms for Percontrol. Our choice for an AI mechanism dwelt on neural networks, while routing mechanisms were TCP/IP and ZigBee. These choices are supported by published works in routing protocols [18], artificial intelligence [43], and comparison and use of neural networks [27, 45].

The neural network loads the entire history of a device being handled within the environment, and uses its historical data as training, in order to identify decision patterns that were assumed in a recent past. Considering our air-conditioning example, these patterns include the temperature that each user wants for a certain environment and what temperature was actually used when all users were taken into consideration. This type of analysis is crucial for the network’s decision—making process. Fig. 7 presents part of the source code used to define the desired and assumed temperatures. These values are fixed for

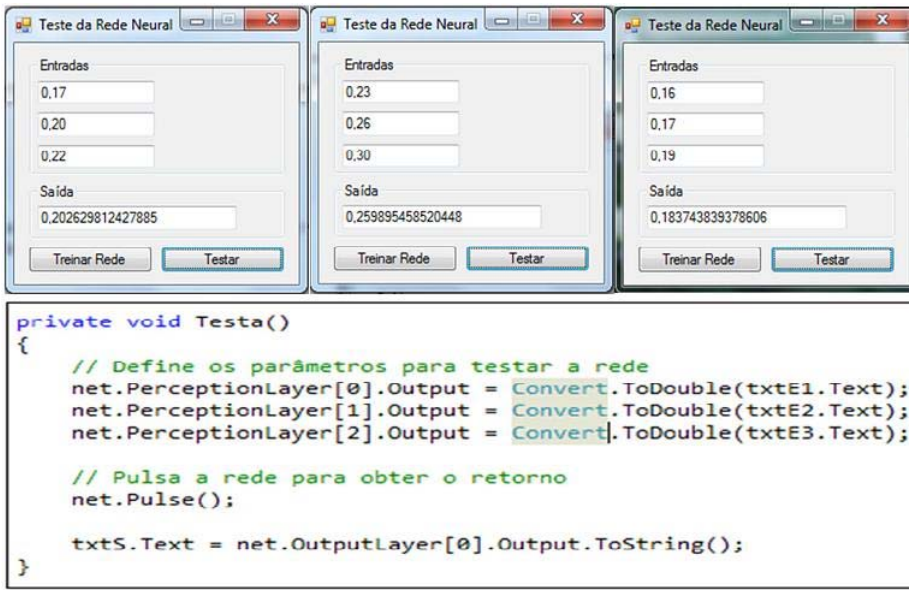

Fig. 7 Part of the neural network's source code.

testing, but in a real scenario they are fetched from a data base or an archive. The code shown in Fig. 7 is used to train the neural network. After training phase the next step is to test the network to determine if it is well-suited to solve the problem of finding the ideal temperature using past event data; in order to perform the testing, a graphical interface was developed. The interface receives the values for current data and returns the ideal temperature estimation, as shown in Fig. 8.

After the neural network's training, we could identify the network's response time after a user enters the environment, as shown in Fig. 9.

The Fig. 9 presents response times of the ANN for the cases with 1, 3 and 10 distinct users identified by the pervasive environment, where X axis represents the number of users and Y axis represents the elapsed time. For a single user, the ANN took 3 seconds to process the information contained in the user's profile, returning an average temperature with a value equal to the one defined by the user (since it is just a single person). For three users, the neural network took 5

```
// Inicializa com
//   1 randon
//   3 perception neurons (entradas)
//   6 hidden layer neurons
//   1 output neuron (saída)
net.Initialize(1, 3, 6, 1);

// Taxa de aprendizagem
net.LearningRate = 3;

// Iterações
iterations = 10000;

// Treina a rede
net.Train(input, output, TrainingType.BackPropogation, iterations);
}
```

Fig. 8 Training the ANN.

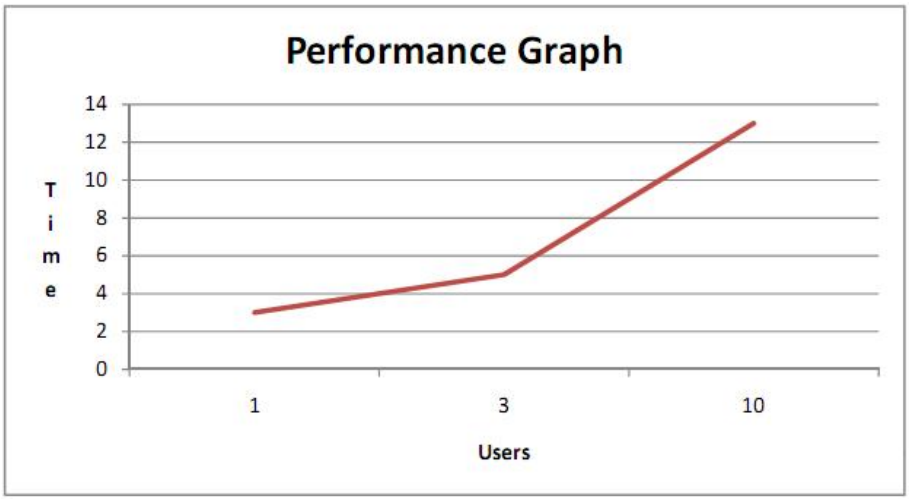

Fig. 9 Neural network's performance and response time.

seconds to respond, and for ten users it took 13 seconds. In Fig. 10, a screenshot shows information on the users identified by the system, as well as on the devices associated with them.

To perform the identification of different environments, we used an Arduino Duemilanove [40]. It is a microcontroller board based on ATmega328, with 14 digital input/output pins, 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. The board contains the necessary assets to support the microcontroller and its use is as simple as connecting it to a computer with a USB cable or powering it with an AC-to-DC adapter or battery.

With this board, it was possible to detect devices via Bluetooth, Wi-Fi and, after being integrated with an appropriate card reader, RFID. The reader fetched a RFID card's serial number that can be cross matched with the user's registration on the data base. For this purpose, a RFID card reader model YHY502CTG was used in conjunction with the Arduino board. After obtaining the necessary application data and performing

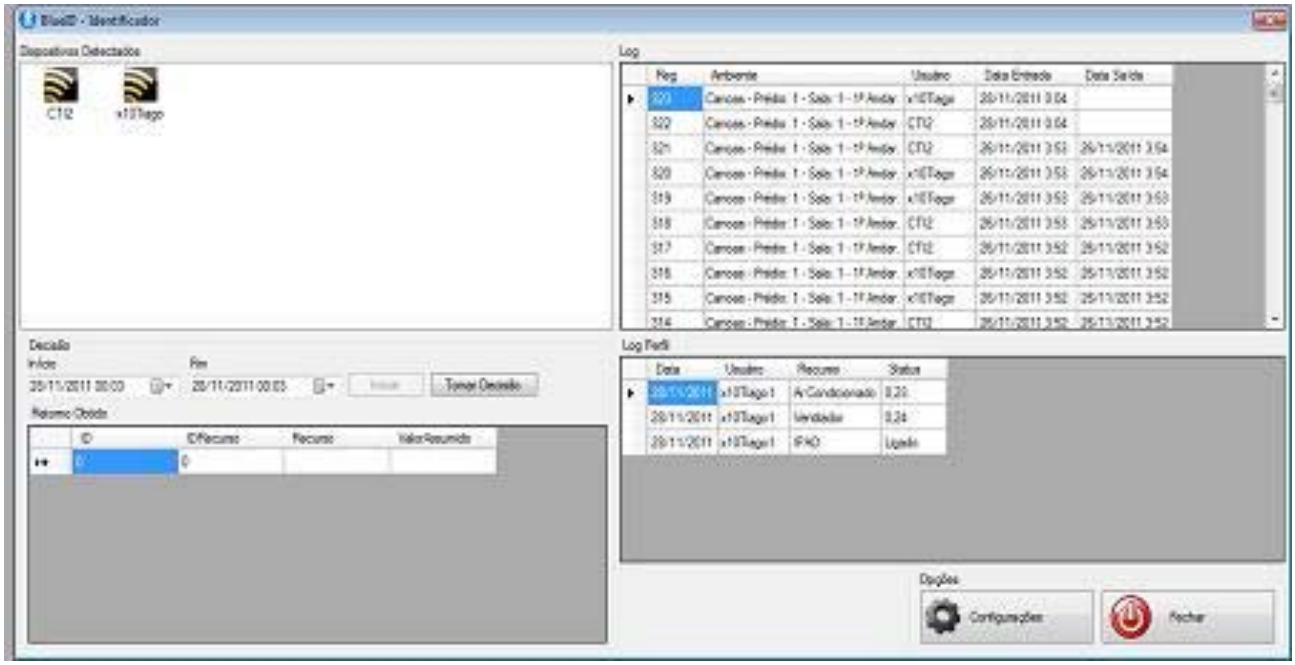

Fig. 10 User identification screen.

Fig. 11 Validation equipment.

the necessary adjustments, the system was validated using a didactic MultiPIC development Kit which possesses its own internal programmer. The didactic development Kit was connected to a stepper motor that simulated a ventilator. The stepper motor can be put in action with different speeds; initially, we defined 3 different speeds that corresponded to 3 different profiles. Fig. 11 shows a picture of the assembled device.

The software used for simulating a ventilator with 3 speeds and controlling the board and stepper motor was developed in C language using the development software from Microchip MTLab, and transferred to the microcontroller with the IC-Prog 1.06C, the software used to compile the source code onto the MultiPIC Kit's processor.

On the performed tests, the ANNs computed the average temperature from the user profiles and used current environmental information from the SunSpot sensors to correctly manage the ventilation system. From these tests, we conclude that Percontrol managed the pervasive environment in a satisfactory manner and that the primary objectives of this research were met, although there is still much room for improvements.

6. Conclusion and Future Work

There are several papers that cover one or two of the taxonomic topics related to pervasive and ubiquitous computing, but few describe how in the future they will give priority to the treatment of privacy in pervasive environments and not just to the elements that surround it. A primary goal of our study was to identify the related work to the management and control of the privacy in pervasive and ubiquitous context. According to the current literature, the main researches on control and management of privacy are about communication, applications and services, user and devices. This paper advances the state of art about pervasive environments and proposes a generic model to control and manage the privacy on these scenarios. The main focus of the model is the environment instead of only the users and their devices. A prototype was developed to test to validate the generic model of privacy. The results confirmed the viability of device detection with Wi-Fi, Bluetooth and RFID and an improvement over previous Percontrol versions. Nevertheless, there is still some latency in registering new devices on the system, which may be reduced by further adjustments of the parameters sent to the ANNs. This work represents a significant contribution since it covers different areas and technologies within pervasive computation. In the future work, several parameters and definitions will be implemented and tested, new models of privacy control for users, devices and environments will be considered.

Acknowledgments

This work was developed with the support of the CNPq Brasil (Conselho de desenvolvimento Científico e Tecnológico), with resources provided by the SWE Program (Doutorado Sanduíche Programa Brasileiro Ciência sem Fronteiras) and by CAPES in the post-doctoral. Also, it is supported by the Project "Pesquisadores na Empresa Projeto GVWISE-GVDASA-UNISINOS-CNPq Brasil", the UFRGS (Federal University of Rio Grande do Sul), the SAPO/Portugal Telecom and the Department of Informatics Engineering of the University of Coimbra,

Portugal.

References

[1] M. Weiser, The Computer for the Twenty-First Century, Scientific American (1991) 94-10.

[2] M. Satyanarayanan, Pervasive computing: Vision and challenges, IEEE Personal Communications 8 (2001) 10-17.

[3] D. Saha, A. Mukherjee, Pervasive computing: A paradigm for the 21st century, Computer 36 (2003) 25-31.

[4] RFID Journal, http://www.rfidjournal.com (accessed April 2012).

[5] Wireless Sensor Network [Online], Scientific Research, http://www.scirp.org/journal/wsn (accessed May 2012).

[6] S. Warren, L. Brandeis, The right to privacy, Harward Law Review 4 (1890) 193-220.

[7] V.R.Q. Leithardt, D. Nunes, A.G.M Rossetto, C.O. Rolim, C.F.R. Geyer, J.S. Silva, Privacy Management Solution in Ubisuitous Environments Using Percontrol, Journal of Ubiquitous Systems and Pervasive Networks (JUSPN) 5 (2014) 21-28.

[8] J. Ye, L. Coyle, S. Dobson, P. Nixon, Ontology-based models in pervasive computing systems, Knowl. Eng. Rev. 22 (2007) 315-347.

[9] E. Moschetta, R.S. Antunes, M.P. Barcellos, More flexible degrees of collaboration, security and privacy in service discovery in ubiquitous environments, in: XXVI Brazilian Symposium on Computer Networks and Distributed Systems (SBRC 2008), Rio de Janeiro, Brazil, May 26-30, 2008.

[10] M. Ros, M. D'Souza, A. Postula, I. MacColl, Wireless outdoor personal area network using adaptive inquiry scanning for location-based services, Personal and Ubiquitous Computing 17 (2013) 387-397.

[11] J. Krumm, A survey of computational location privacy, Personal Ubiquitous Comput. 13 (2009) 391-399.

[12] A.R. Beresford, F. Stajano, Location privacy in pervasive computing, IEEE Pervasive Computing 2 (2003) 46-55.

[13] A. Gorlach, A. Heinemann, W.W. Terpstra, Survey on location privacy in pervasive computing, in: Privacy, Security and Trust within the Context of Pervasive Computing, The Kluwer International Series in Engineering and Computer Science, 2004, pp. 23-34.

[14] D. Lupiana, C. O'Driscoll, F. Mtenzi, Taxonomy for ubiquitous computing environments, in: First International Conference on Networked Digital Technologies, Ostrava, July 2009, pp. 469-475.

[15] A. Patwardhan, V. Korolev, L. Kagal, A. Joshi, Enforcing policies in pervasive environments, in: The First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, Aug. 2004, pp. 299-308.

[16] F. Zhu, M.W. Mutka, L.M. Ni, Service discovery in pervasive computing environments, IEEE Pervasive Computing 4 (2005) 81-90.

[17] K.E. Defrawy, G. Tsudik, Privacy-Preserving location-based on-demand routing in MANETs, IEEE Journal on Selected Areas in Communications 29 (2011) 1926-1934.

[18] R.Q.V Leithardt, C.F.R. Geyer, J.E.R. Tavares, Comparative study of routing protocols used in wireless sensor networks, in: IV Congresso da Academia Trinacional da Fronteira, Foz do Iguaçu, Brazil, 2009.

[19] R.C. Mickunas, J. Al-Muhtadi, P. Naldurg, G. Sampemane, M. Dennis, Towards Security and Privacy for Pervasive Computing [Online], http://srg.cs.uiuc.edu/gaia/papers/ towards-percomp-security.pdf (accessed February 2012).

[20] V.C.K. Sobral, J.B. Mangueira Mangueira, Especificando Privacidade em Ambientes de Computação Ubíqua. http://www.inf.ufsc.br/~bosco/ensino/ine6406 (accessed November 2011).

[21] C.O. Rolim, V.R.Q. Leithardt, A.G. Rossetto, T.F.M. dos Santos, A.M. Souza, C.F.R. Geyer, Six degrees of separation to improve routing in opportunistic networks, International Journal of UbiComp 4 (2013) 11-22.

[22] V. Pereira, J.S. Silva, J. Granjal, R. Silva, E. Monteiro, Q. Pan, A Taxonomy of Wireless Sensor Networks with QoS, in: 2011 4th IFIP International Conference on New Technologies, Mobility and Security (NTMS), Paris, Feb. 2011, pp. 1-4.

[23] N. Li, N. Zhang, S.K. Das, B. Thuraisingham, Privacy preservation in wireless sensor networks: A state-of-the-art survey, Ad Hoc Networks 7 (2009) 1501-1514.

[24] L. Kagal, T. Finin, A. Joshi, Trust-based security in pervasive computing environments, Computer 34 (2001) 154-157.

[25] K. Henricksen, R. Wishart, T. McFadden, J. Indulska, Extending context models for privacy in pervasive computing environments, in: Third IEEE International Conference on Pervasive Computing and Communications Workshops, Kauai Island, March 2005, pp. 20-24.

[26] V.R.Q. Leithardt, G.A. Borges, I.M. Carrera, A.G.M. Rossetto, C.O. Rolim, D. Nunes, S.J. Silva, C.F.R. Geyer, Mobile architecture for identifying users in ubiquitous environments focused on Percontrol, in: The Seventh International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies, UBICOMM 2013, Porto, 2013. pp. 145-151.

[27] C.O. Rolim, F. Schubert, A.G.M. Rossetto, V.R.Q.

Leithardt, C.F.R. Geyer, C. Westphall, Comparison of a multi output adaptative Neuro-Fuzzy inference system (manfis) and Multi Layer Perceptron (mlp) in cloud computing, in: 29th Brazilian Symposium on Computer Networks and Distributed Systems, Paris, July 25-27, 2012.

[28] K. Shankar, L. Camp, K. Connelly, L. Huber, Aging, privacy, and home-based computing: Designing for privacy, IEEE Pervasive Computing 11 (2011) 46-54.

[29] J.T. Lehikoinen, J. Lehikoinen, P. Huuskonen, Understanding privacy regulation in ubicomp interactions, Personal Ubiquitous Comput. 12 (2008) 543-553.

[30] G. Iachello, J. Hong, End-user privacy in human-computer interaction, Foundation and Trends in Humn-Computer Interaction 1 (2007) 1-137.

[31] J. Bardram, R. Kjær, M. Pedersen, Context-Aware User Authentication—Supporting Proximity-Based Login in Pervasive Computing, in: UbiComp, Seattle, 2003, pp. 107-123.

[32] V.R.Q. Leithardt, C.O. Rolim, A.G.M. Rossetto, C.F.R. Geyer, M.A.R. Dantas, J.S. Silva, D. Nunes, Percontrol: A pervasive system for educational environments, in: 2012 International Conference on Computing, Networking and Communications (ICNC), Maui, Jan. 30-Feb. 2, 2012, pp. 131-136.

[33] A.K. Dey, Providing architectural support for building context-aware applications, Ph.D. Thesis, College of Computing, Georgia Institute of Technology, Atlanta, December 2000.

[34] Europe's Information Society [Online]. http://ec.europa.eu/information/society/policy (accessed March 2012).

[35] R. Babbitt, J. Wong, C.K. Chang, Towards the modeling of personal privacy in ubiquitous computing environments, in: 31st Annual IEEE International Computer Software and Applications Conference, Beijing, July 24-27, 2007, pp. 695-699.

[36] Texas Instruments. Texas instruments [Online], http://focus.ti.com/pdfs/wtbu/ti_mid_whitepaper.pdf (accessed December 2011).

[37] R. Silva, V.R.Q. Leithardt, J.S.Silva, C.F.R. Geyer, J. Rodrigues, F. Boavida, A comparison of approaches to node and service discovery in 6lowPAN wireless sensor networks, in: Proceedings of the 5th ACM Symposium on QoS and Security For Wireless and Mobile Networks, New York, Oct. 2009, pp. 44-49.

[38] N. Li, N. Zhang, S. K. Das, B.M. Thuraisingham, Privacy preservation in wireless sensor networks: A state-of-the-art survey, Ad Hoc Networks 7 (2009) 1501-1514.

[39] Sun spot world, Oracle Labs, http://www.sunspotworld.com/ (accessed May 2013).

[40] Arduino brasil, http://www.arduino.com.br/ (accessed April 2013).

[41] A.A. Loureiro, J.M.S Nogueira, L.B. Ruiz, R.A. Mini de Freitas, E.F. Nakamura, C.M.S. Figueiredo, Wireless sensor networks, in: Brazilian Symposium on Computer Networks (SBRC), May, Porto Alegre, 2003, pp. 179-226.

[42] A.D.P. Braga, T.B. Ludemir, A.C.P.F. Carvalho, Artificial Neural Networks: Theory and Applications, LTC Publisher, Rio de Janeiro, Brazil, 2007.

[43] O. Ludwig Jr., C.M.E. Montgomery, Neural networks: Fundamentals and applications in C program, Rio de Janeiro: Editora Moderna LTDA Science, 2007.

[44] C.O. Rolim, A.G. M. Rossetto, V.R.Q. Leithardt, C.F.R. Geyer, Analysis of a Hybrid Neural Network as a basis for prediction mechanism Situation, in: Congress of the Brazilian Computer Society (CSBC 2012) Symposium on Pervasive Ubiquitous Computing, Brazil, 2012.

[45] R. Silva, J. Sá Silva, C. Geyer, V. Leithard, F. Boavida, Use of GPS and 6lowpan in mobile multi-sink wireless sensor networks—issues and feasibility, in: 8th International Information and telecommunication Technologies Symposium, IEEE R9 Latim America, 2009, pp. 154-160.