Privacy Infusion in Ubiquitous Computing

Gautham Pallapa, Mohan Kumar, and Sajal K. Das
Department of Computer Science and Engineering
University of Texas, Arlington
Arlington, Texas 76019
Email: {pallapa, kumar, das}@cse.uta.edu

Abstract— **In recent years, ubiquitous computing applications span such areas as telemedicine, banking, and transportation that require user privacy protection. The realization of context-aware ubiquitous computing exasperates existing privacy concerns. Ubiquitous computing applications demand new privacy enhancing technologies for the information and communication environments where the users are equipped with flexible and portable applications to support the capture, communication, recall, organization and reuse of diverse information. In this paper, we develop a novel scheme for the infusion of privacy into context-aware ubiquitous computing. We present Precision, a system for privacy enhanced context-aware information fusion in ubiquitous computing environments. In our scheme, privacy is defined as a set of parameters encapsulated in composite data entities called privons, through which, we aim at infusing privacy into Precision. We evaluate our proposed scheme through real interactions in implementation of privons.**

Keywords: Privacy, Context, Information Fusion, Middleware, Ubiquitous Computing

I. Introduction

Ubiquitous computing, in today's world, usually implies embedding the technology unobtrusively within all manner of everyday computers or appliances that can potentially transmit and receive information among each other [1]. In light of these developments, it is complacent to assume that social and organizational controls over accessibility of personal information are sufficient [2], or that intrusions into privacy will ultimately become acceptable when traded against potential benefits [3], [4]. Such an assumption could leave individual users with a heavy burden of responsibility to ensure that they do not, even inadvertently, intrude on others. It also leaves them with limited control over their own privacy.

Some of the ubiquitous computing (ubicomp) systems have no mechanism for people to reflect upon the system, to see how they and their information affect, contribute, interact, or participate in the system [5]. It is also imperative that users are made aware of the limitations and constraints of the system. For example, a kitchen system that helps a person maintain groceries in the house will have sensing limitations, which may be dynamic, based on sensors and situations. Effective use of this tool requires understanding what the sensing system missed so that a person can compensate. As ubicomp systems rely on implicit sensing that is naturally ambiguous or error-prone, designers must help users comprehend the limitations of the system [6].

A frequent definition of information privacy is "*the claim of individuals, groups or institutions to determine for themselves when, how, and to what extent information about them is communicated to others*" [7]. Despite this definition, "not sharing information" is a fundamental aspect of privacy. The real privacy concerns people have today revolves around who has access to what information rather than what information is collected [8]. The ethics of privacy can emanate from the fact that *"knowledge is power"*. It is essential to have fine-grained control over privacy-sensitive information.

One of the key concepts of ubiquitous computing is empowering technology to create a convenient environment surrounding the user which merges physical and computational infrastructures into an integrated information oriented habitat [9]. One of the challenges in deploying ubiquitous computing services on a significant scale is making adequate provision for handling personal privacy. The objectives of privacy and ubiquity are polar in nature. On the one hand, to ensure ubiquity, the applications are tuned to gather more information about the user to perform their tasks better. On the other hand, the more the application knows about the user, the greater is the threat to the user's privacy [10]. To develop a scheme that infuses privacy and the context of an event in the middleware of the ubiquitous computing framework is a significant challenge.

Realization of reciprocity in ubiquitous computing environments is complex because of heterogeneity of devices, their capabilities, and heterogeneous communication protocols used. To simplify application development, a middleware developed for a ubiquitous computing environment has to be *service-oriented* and should be able to endure limited device capabilities [11]. Fault-tolerance, integration, reconfiguration and usability are some of the innate characteristics of such a middleware. While these characteristics are important in terms of system deployment, the focus should be more on usability and inter-operability. Added to these challenges is the seamless handling of the dynamic nature of ubicomp systems. Unfortunately, accommodating all these features in any system introduce many risks, privacy being one of them.

Acknowledging this complexity in privacy, we introduce a composite entity called *"privon"* to encapsulate related information based on their privacy-sensitivity levels. A privon is envisioned to allow users (1) to understand how their personal information may be used by others, and (2) to understand how they and their information participate in the system. In our scheme, we use privons to infuse privacy into our context-aware framework. In the following sections, we discuss the concept of privons and also present Precision, a system for privacy enhanced context-aware information fusion in ubiquitous computing environments, and discuss the incorporation of privons to Precision.

II. Motivation and Background

Privacy is the ability of an individual or group to keep related information, their social and behavioral patterns out of public view, or to control the flow of information about themselves. Traditionally, privacy enhancing technologies [12] have been thought as tools for hiding, obfuscating, and controlling disclosure. But in terms of an overall approach to privacy management, it is necessary to think about how technology can be used to create visibility and awareness of information security practices. The Health Insurance Portability and Accountability Act (HIPAA) Privacy Rule standards [13] address the use and disclosure of individuals health information. A major goal of the HIPAA Privacy Rule is to assure that individuals health information is properly protected while allowing the flow of health information needed to provide and promote high quality health care and to protect the public's health and well being.

The perception of privacy in the aspects of the information receiver, user and sensitivity in terms of design was investigated in [14]. Tentori et al. [15], introduced the concept of Quality of Privacy to address the tradeoff between the services provided by a ubiquitous environment and the cost that the users might need to pay in regard to privacy. Though this scheme incorporated context-aware communication, the system was not suggestive and depended on the user's knowledge of information sharing. Confab [16] offers a framework where personal information is captured, stored and processed on the end-user's computer as much has possible. Though this addresses the high-level requirements of decentralized architecture and plausible deniability, and offers a larger amount of choice and control than previous systems, the system is not obstrusive in nature. A scheme that alleviates loss of privacy without forbidding the use of trust is described in [17]. A study of the relationship between context and privacy was made in [18]. [19] presented an architecture for privacy-sensitive ubiquitous computing, where the authors claim that the large majority of work on privacy has tended to focus on providing anonymity rather than considering the many scenarios in everyday life, where people want to share information. Owing to the nature of privacy, it is difficult to design privacy-sensitive ubiquitous applications.

Context is the circumstance under which a device is being used, e.g., the current occupation of the user. Toivonen et al. [20] discuss methods for evaluating context-aware trust for dynamic reconfigurable systems. In information systems, context information is metadata stored in the system about entities (e.g., resources, events, etc) that are related to, but not intrinsically about the entity itself. The line between ordinary metadata and contextual metadata is a blurred one, and often depends on individual perspectives about what counts as intrinsic information. A common deliminator is whether the information originated internally (such as a timestamp generated by the system clock) or externally via some form of sensor (such as the weather at that time).

Designers face many challenges while building a framework for a ubicomp environment. The key challenge is to engineer a framework capable of adapting to such a highly chaotic environment and seamlessly integrate itself with the existing legacy systems. To illustrate these points, we present the following scenario:

Scenario:

John stores his emergency information, auto insurance, and medical records on his PDA. If the PDA detects an accident, it sends a dialog box, requesting John if he needs medical assistance. Upon receiving no response, the PDA places a call to the Emergency Medical Services (EMS), giving the location of the user. When the Emergency Medical Technicians (EMTs) arrive, they request all the devices of the user to transmit data about the user. When the PDA checks that the request was issued by an Emergency Medical Technician, it transmits the personal and medical details, which the EMT aggregates with the vital signs and health monitoring data and transmits it to the Emergency Room prior to arrival, facilitating the doctors in the ER to prepare for the victim. A case sheet is generated with the patient details and the doctor on call, Dr. Alice is informed of the status of the new case. Suppose, Dr. Alice wishes to discuss John's case with Dr. Bob, she messages him and asks if he is free to discuss the case. On receiving Dr. Bob's response, Dr. Alice's laptop decreases the privacy level of the session since the data is being sent to a peer (doctor). It then collects all the related patient information, checks the target device and makes necessary changes to match Dr. Bob's device and his privacy settings. Concurrently, a video conference session is set up between the doctors. When the session ends, Dr. Alice's privacy level is set back to the default setting.

The common vulnerabilities encountered while developing such a ubicomp system can be categorized into *flow-based*, or *process-based*. The former is the inability to associate relevant information and lack of transparency, and the latter deals with issues such as configuration being given importance over action, and the granularity of the system incorporating social variations. These two categories are interwoven, but demarcating them can help designers in analyzing them. Some of the current problems involved in developing a privacy-sensitive ubiquitous computing framework are:

A. Inabibility to associate relevant information

A ubicomp environment is a challenge to designers of middleware. It is difficult to encompass the myriad information flows that exist in such a chaotic environment. Many of the middleware are designed as an Event-Condition-Action (ECA) approach [21], [22], [23]. Importance has to be given to the relevance of data pertaining to a session. It would be more advantageous taking a user's behavior as an entity and deriving work flows from it, rather than considering events as a basal unit [24]. The middleware should be able to predict the information required for a particular service. For example, a request for a user's contact details should include email address, address, and phone number. It is redundant to have multiple requests for each piece of information.

B. Lack of transparency in authentic information

In human-computer interaction, computer transparency is an aspect of user friendliness which relieves the user of the need to worry about technical details. When there is a large gap between user perception and actual authentic information, the system is failing in representation of information. Information transparency changes behavior [25], and there have been some efforts in the field of privacy enhancing technologies that help create transparency of information security practices.

One problem area to be tackled is that of sharing and distributing information between users, i.e., not only between all participants in a single application such as a conference, but also across different applications, for e.g., information retrieval. This makes the need of information brokers imperative. CORBA Component Model (CCM), an extension of "language independent Enterprise Java Beans (EJB)" [26], is a broker oriented architecture. By moving the implementation of these services from the software components to a component container, the complexity of the components is dramatically reduced [27]. One drawback of CCM is the lack of provision for tackling the issue of disconnected processes, which is rampant in a ubiquitous computing environment [28].

C. Configuration superseding action

Designers constantly include huge configuration steps for incorporating privacy into the model. This would be necessary for making the system robust, but deters the user from using the system effectively. Web services [29], [30] aims at promoting a modular, interoperable service layer on top of the existing Internet software [31], but lacks consistent management and are tightly bound to the simple object access protocol (SOAP) which constrains compliance to various ubiquitous computing protocols. Jini [32], is a service oriented architecture that defines a programming model which both exploits and extends Java technology to build adaptive network systems that are scalable, evolvable and flexible as typically required in dynamic computing environments. Jini, however, assumes that mobile devices are only service customers which is not the case. We aim at reducing the task of user configuration by introducing classification of information based on privacy levels.

D. Granularity of the system incorporating social variations

The ubiquitous framework should be able to predict the privacy level of the session based of peer bonding and organizational hierarchy. At the same time, it should allow the user to set privacy levels to other individuals based on their social interaction. Since it is difficult to define privacy, we considered it beneficial to incorporate a privacy slider to effectively depict the granularity of user interpretation. In a social environment, maladroit situations, such as denial of a service or a request for information, have to be handled gracefully.

These conundrums that are constantly faced while developing middleware frameworks for ubicomp environments form the motivation for this work. Security and privacy have an implicit relationship. An elementary level of security is imperative while helping people manage their personal privacy. Since, in many scenarios of a ubiquitous computing environment, the users are not necessarily adversaries, and usually know each other's identities, the uncertainty would be less and hence we adapt a privacy risk model rather than a threat model. Social and organizational context should also be taken into consideration while developing a framework for the environment [33].

The scenario described, though common in real life, involves modifications to the system based upon various context elements such as location, device properties, etc., and also on the social interactions of the users. In this paper, we focus on patterns of personal information disclosure and privacy implications associated with user interactions in a ubiquitous computing environment. We propose a ubiquitous framework which modifies the presentation of data based upon the device and user profiles.

III. The Proposed Scheme

In this section, we introduce the concept of privons as composite entities, comprising of privacy levels, data, and services, developed for handling privacy. We also discuss the architecture of Precision, our privacy enhanced context-aware information fusion system for ubiquitous computing environments. We have adapted a weighted scheme to incorporate privacy in the privons stored, aiming at introduction of learning mechanisms for privacy management to reduce obstrusiveness in Precision. We generate privons for all sessions invoked in Precision and transfer information through these entities.

A. Privons

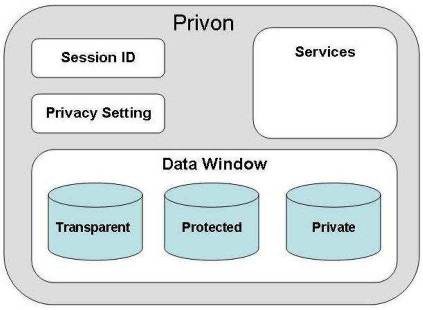


Fig. 1. Structure of a Privon

Privacy is one of the areas of computing with tradeoffs, and is a difficult, yet necessary, design issue in ubiquitous computing environments. Context and social nuances contribute to making privacy for a user, making it a meta-property. In order to process the abstract nature of personal privacy and convert it into tangible issues, we introduce the concept of privons.

The conceptual structure of a privon is shown in Figure 1. A privon consists of a session id, privacy settings, services and a data window. Data in a privon can exist in on the the following three classes: transparent, protected, or private. A data element belongs to one of these based upon the privacy level of the session, set by either the user or based on the session characteristics. The data window is a data repository for these three classes of data. Transparent data is visible to all devices in the network and constitutes the main portion of the data window. Depending on the privacy settings, the data window can be updated incorporating data elements from protected or private data. Based on the user configuration, elements of transparent data can be shifted to any of the other classes.

Services pertaining to retrieval and aggregation of data are also included in a privon. For example, with regard to our scenario described in Section II, request for pertinent data about a victim would be a service included in the privon issued by the EMTs. Aggregation of all medical data and outputs from the various health monitors would be another example of a service in a privon.

B. Precision

We have developed a Privacy enhanced Context Aware Information Fusion framework for ubiquitous computing environments called Precision to handle personal privacy of a user in highly dynamic environments. Figure 2 shows the architecture of Precision.

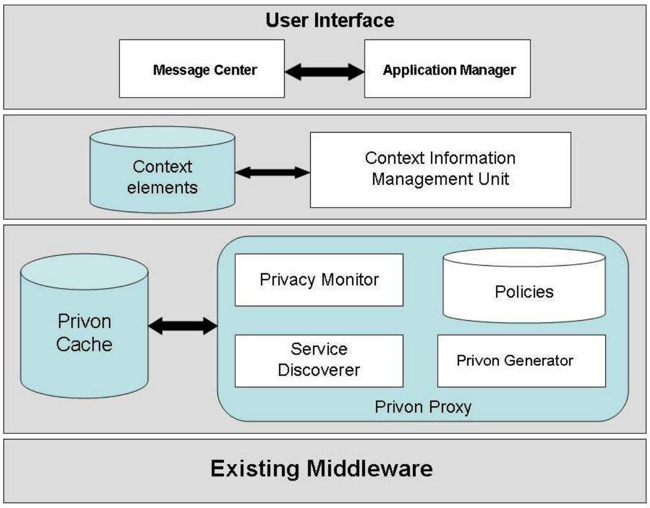


Fig. 2. Proposed framework of Precision

The user interface consists of a message center and an application manager. The application manager runs the various services on the device and also manages the requests received from, and responses sent to, the user. These are displayed using the message center to increase the usability of the system. The message center is designed on the lines of a chat client, and is used to capture the intent of the user, and forward it to the application manager.

The application manager sends the requests and receives responses from the Context Information Management Unit (CIMU). The CIMU is responsible for obtaining the context elements associated to the data elements being utilized in a particular session. Weights are assigned to the data elements based on the pertinence to the associated context elements and these weights are sent to the privon proxy for pruning, based on the privacy policies set by the user.

The privon proxy consists of three units: *privacy monitor, service discoverer,* and *privon generator*. All the policies and user configurations are stored in the *policies* cache and the policy relevant to the current session is accessed by the privacy monitor. The weighted data elements, generated by the CIMU, are modified based on the policies and the new weighted data elements are segregated into the different classes of data and the new privon is then generated by the privon generator, and dispatched to the recipient. The generated privon is also stored in the privon cache.

The functions of the service discoverer are twofold. When a request privon is to be generated, the relevant service is sent to the privon generator for incorporation. Additionally, other services could also be sent to the privon generator. When a privon is received, the service discoverer scans the services area of the privon and based on the privacy settings, deploys the requested service. The functions of the service discover could also be extended to resource discovery and agent deployment, but we have considered information request as the predominant service for this work.

C. Privon Generation

Privons are generated for a session established between two users in the ubiquitous computing environment. When a user initiates a session to exchange information, the application manager allots a session id, and the list of requested data elements is obtained through the message center from the user. The privacy level or privacy coefficient of the session α, is set based on the user settings and policy of the session and the data elements are classified. The CIMU extracts the context elements corresponding to the data elements and passes them to the privon proxy. The data window of the privon is increased to include the requested data elements. The privon generator includes the *requestInformation* service, and creates the request privon. The privacy settings of the session are checked by the privacy monitor and added to the privon. This privon is then dispatched to the recipient, and stored in the privon cache. The overview of privon generation is depicted in Figure 3.

D. Classification of data

Let U be a set $\{u_1, u_2, \ldots, u_n\}$ which contains all the data elements of the user, and $C = \{c_1, c_2, \ldots, c_m\}$ be a set of all context elements . When a user sends information to a request, the input from the user is broken down, and the corresponding data elements form a set U^*. It is intuitive that $U^* \subseteq U$. The context elements attributed to the corresponding entries in U^* are retrieved from C and form a subset C^*. The probability

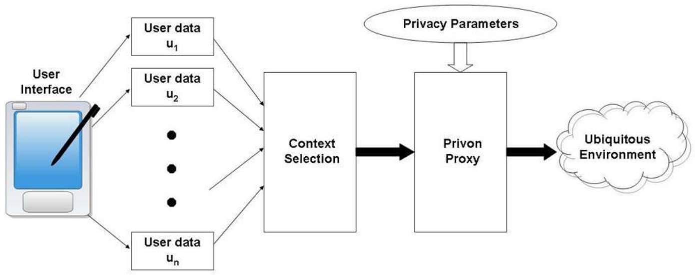


Fig. 3. Overview of privon generation

that a particular context element c_i is chosen, given that data element u_j is accessed is given by

$$P(c_i^*) = P(c_i|u_j), \qquad i = 1, \dots, m, j = 1, \dots, n \quad (1)$$

All k context elements related to data element u_i are considered, and the weight w_i of u_i is given by

$$w_i = \frac{\sum_{j=1}^{k} P(c_j^*)}{k} \quad (2)$$

Thresholds are set for distinguishing the different classes of data, based on the weights of data elements. For our scheme, we have considered three groups C_T, C_D, and C_P corresponding to *transparent, protected,* and *private* data.

We define two functions $\text{Inc}_\delta(w)$ and $\text{Dec}_\delta(w)$ which increment or decrement the value of w by δ, thereby, increasing or decreasing the privacy setting for w. These two functions are of type $[0,1] \rightarrow [0,1]$ such that given a weight w, they return an incremented or decremented weight, $w^{'}$, respectively. We assume that these functions [20] satisfy the following two properties:

- $\text{Inc}_\delta(\text{Dec}_\delta(w)) = w$ and $\text{Dec}_\delta(\text{Inc}_\delta(w)) = w$, i.e., they are mutually commutative.
- The two functions are order-independent with regard to the context elements, i.e., $f_\delta(g_{\delta'}(w)) = g_{\delta^1}(f_\delta(w))$ where $f, g \epsilon \{Inc, Dec\}$

We choose the following functions, which satisfy the above properties, to calculate the adjustments:

$$Inc_\delta(w) = \sqrt[\delta]{w} \quad (3)$$

$$Dec_\delta(w) = w^\delta \quad (4)$$

When the context elements are selected, we apply these functions to the weights assigned to the elements in C^* and $C - C^*$, such that they satisfy Equation 8

$$\hat{w}_i = \text{Inc}_\delta(c_i), \qquad c_i \in C^* \quad (5)$$

$$\hat{w}_i = \text{Dec}_\delta(c_i), \qquad c_i \in C - C^* \quad (6)$$

Weight thresholds are set for membership to the three classes of data, and these classes are dynamically organized based on the privacy settings of the current session, satisfying the condition

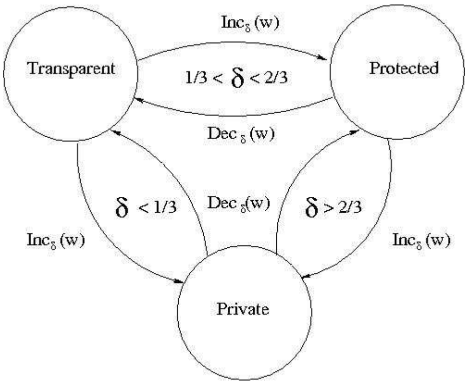


Fig. 4. Data Classification

$$\sum_{i=1}^{n} \hat{w}_i P(c_i) \leq \alpha \quad (7)$$

where, n is the number of data elements in the data window of the generated privon. The ranges of δ which cause transition of the data elements from one class to the other is shown in Figure 4. We have set the thresholds such that the same range of δ can increase or decrease the privacy level of a data element based on the type of weight function applied. For instance, a data element currently belonging to the *transparent* class, makes a transition to *protected* by applying $\text{Inc}_\delta(w)$, where $\frac{1}{3} < \delta < \frac{2}{3}$. On the other hand, if $\delta < \frac{1}{3}$, the data element becomes *private*.

Let (ε, P) be a joint probability space with $\varepsilon = \varepsilon_1 \times \ldots \times \varepsilon_n$, and joint probability P. Let $\pi(c_i)$ be the parents of c_i and $A(c_i)$ be the nondescendant of $\pi(c_i)$[1]. The resulting joint probability over context, which forms the random variable, forms the weight for a data element, and can be expressed as

$$\hat{w}_i = \prod_{i=1}^{m} P(c_i|\pi(c_i)) \quad (8)$$

The data elements form a directed acyclic graph (DAG) of $G = (U, E)$ with arcs E, where U_i is a projection onto ε_i as shown in Figure 5.

E. Storage of the prior sessions

After each session, the data elements that are used can be stored in the privon cache. This consists of a database of the various data elements that were used in the session. Since each data element is associated with a corresponding weight, it would suffice for us to store the weights, rather than the data elements themselves. Each entry consists of the format $T =< S_{id}, \alpha, W >$, where S_{id} is the session id generated, α is the privacy coefficient for the session, and $W = \{w_1, w_2, \ldots, w_k\}$ for the k elements in U^*.

[1] We envision that $\pi(c_i), A(c_i)$, and the directed acyclic graph of data elements generated would be used in the learning mechanism, anticipated as a future work, in Precision

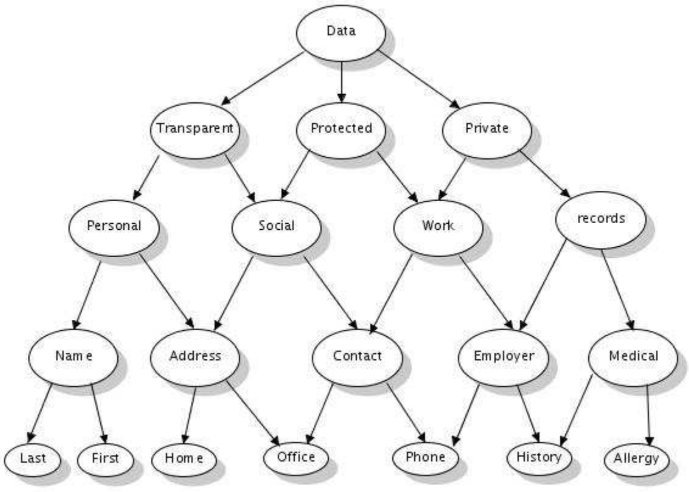


Fig. 5. Directed Acyclic Graph of data elements generated by Precision

F. Privacy Suggestions

When selected data is transferred to a requested device, the privon generator first checks with the entries in the privon cache for any prior instance of a similar nature. Based upon α, the entries in the cache are scanned, and in the presence of a similar tuple, the created privon is compared with the cache entry. In case of dissimilarity, the privon generator generates a message to the user, suggesting modifications to the data transparency. The user can either accept the modifications or override the suggestions. In the event of an override, the new privon settings are now entered into the privon cache. The action of privacy suggestion is described in Algorithm 1. When a similar situation occurs in a future session, both the related entries are suggested to the user. After multiple sessions, the least retrieved entry is flushed from the cache.

IV. Implementation and Evaluation of the Proposed Scheme

We have evaluated our proposed scheme with the scenario described in Section II. We assume that the user, John has keyed in his personal, social, work and medical information in his PDA. In the event of an accident, the system queries John if he is alright or needs any kind of assistance. When there is no response, the system assumes that the user has been incapacitated in some way and sends a distress call to the Emergency Medical Services (EMS). When the Emergency Medical Technicians (EMTs) arrive, they issue a request privon asking for information about the user.

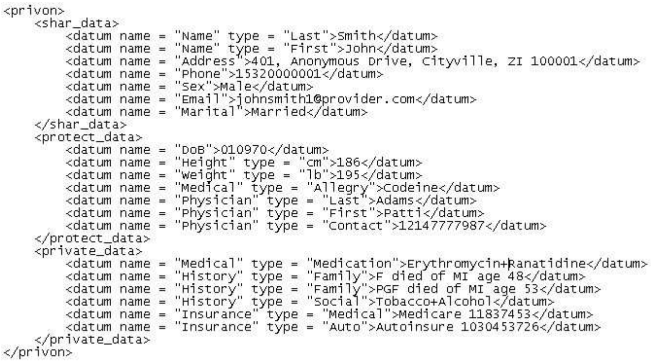
When the instance of Precision on John's PDA receives the request privon, it checks the EMS ID and retrieves the policy for emergency. Though the default privacy setting of a session is low, the system overrides this upon verification of the emergency request. The system constructs the context sets for the three types of data. After this stage, we have $C_T = \{name, phone, address, sex,$ marital status, $email\}$ corresponding to transparent data, $C_D = \{height, weight, DoB\}$ corresponding to protected data, and $C_P = \{$*medical history, medical insurance, physician details*$\}$ corresponding to private data. Upon applying the overriding policy to the context groups, the privacy setting for private and protected data is changed. We then apply the increasing function on the elements in C_P, causing a transition of these elements to C_D.

The transparent and medical data are then aggregated and sent as a response privon to the EMT, depicted in Figure 6. This information is then combined with other vital data, such as the electrocardigram, temperature, still images of the victim, etc., and are sent to the hospital prior to the arrival at the Emergency room in order to facilitate preparation by the medical staff.

Let us now consider the situation where a chat session is

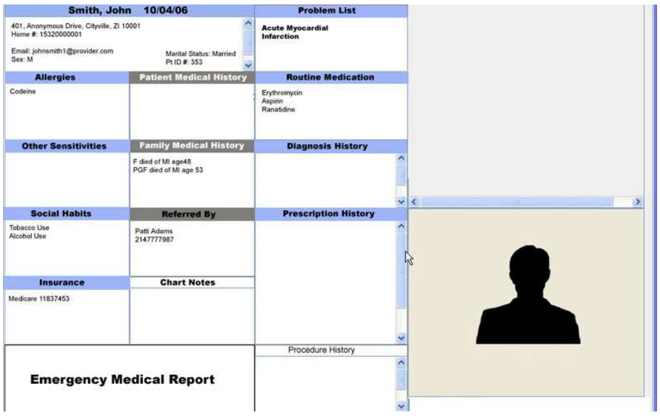
Algorithm 1 Privacy Suggestion

```
    procedure privacySuggest
Require: Current privon tuple
    begin
    for all entries in privon cache do
        if $\alpha_{current} == \alpha_{sessionid}$ then
            retreive table entries for session id
            call $comparePrivon$, input: current tuple, session tuples
        end if
        receive modified tuple
        if modified tuple NOT EQUAL to current tuple then
            call $suggestModify$, input: modified tuple
        end if
    end for
    end
    procedure comparePrivon
Require: current tuple, session tuples
    begin
    for all non-zero weights in current tuple do
        if current weight NOT EQUAL to session weight then
            add session weight to modified tuple
        end if
    end for
    return modified tuple
    end
    procedure suggestModify
Require: modified tuple, session tuples
    begin
    generate message with modified tuple to user
    receive final tuple
    write final tuple to privon cache
    return final tuple
    end
```



```
<privon>
    <shar_data>
        <datum name = "Name" type = "Last">Smith</datum>
        <datum name = "Name" type = "First">John</datum>
        <datum name = "Address">401, Anonymous Drive, Cityville, ZI 100001</datum>
        <datum name = "Phone">1532000001</datum>
        <datum name = "Sex">Male</datum>
        <datum name = "Email">johnsmith1@provider.com</datum>
        <datum name = "Marital">Married</datum>
    </shar_data>
    <protect_data>
        <datum name = "DoB">010970</datum>
        <datum name = "Height" type = "cm">186</datum>
        <datum name = "Weight" type = "lb">195</datum>
        <datum name = "Medical" type = "Allegry">Codeine</datum>
        <datum name = "Physician" type = "Last">Adams</datum>
        <datum name = "Physician" type = "First">Patti</datum>
        <datum name = "Physician" type = "Contact">12147777987</datum>
    </protect_data>
    <private_data>
        <datum name = "Medical" type = "Medication">Erythromycin+Ranatidine</datum>
        <datum name = "History" type = "Family">F died of MI age 48</datum>
        <datum name = "History" type = "Family">PGF died of MI age 53</datum>
        <datum name = "History" type = "Social">Tobacco+Alcohol</datum>
        <datum name = "Insurance" type = "Medical">Medicare 11837453</datum>
        <datum name = "Insurance" type = "Auto">Autoinsure 1030453726</datum>
    </private_data>
</privon>
```

Fig. 6. Dynamically generated privon sent to the EMT



Smith, John 10/04/06

401, Anonymous Drive, Cityville, ZI 10001
Home #: 15320000001

Email: johnsmith1@provider.com
Sex: M

Marital Status: Married
Pt ID #: 353

Problem List: Acute Myocardial Infarction

Allergies: Codeine

Patient Medical History

Routine Medication: Erythromycin
Aspirin
Ranatidine

Other Sensitivities

Family Medical History: F died of MI age48
PGF died of MI age 53

Diagnosis History

Social Habits: Tobacco Use
Alcohol Use

Referred By: Patti Adams
2147777987

Prescription History

Insurance: Medicare 11837453

Chart Notes

Emergency Medical Report

Procedure History

Fig. 7. Patient Chart generated by privons for Scenarios 2 and 3

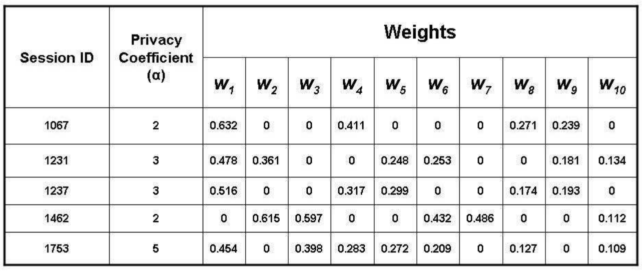
TABLE I
DATA CLASSIFICATION USING PRIVONS

Request of Information	Requested by	Data Classification
User Information	EMT	Transparent
User Location	Emergency Services	Transparent
Medical Information	EMT	Protected
Email address	EMT	Private
Patient Chart	Consigned doctor	Transparent
Patient Chart	Consulting doctor	Protected
User Information	Unknown Entity	Private

initiated by Dr. Alice to Dr. Bob. When Dr. Alice requests for a session with Dr. Bob, a response is awaited and on its receipt from Dr. Bob, the data related to the patient is collected and made transparent. Also the patient chart is accessed and all this information is aggregated with a transparent privacy level as the system recognizes that both the users are peers. The information is then modified to a format compatible with the target device and is sent to Dr. Bob. A video session could also be initiated between the peers to facilitate discussion of the patient's case. Figure 7 shows the output generated by Precision on Dr. Bob's monitor in the form of a patient chart.

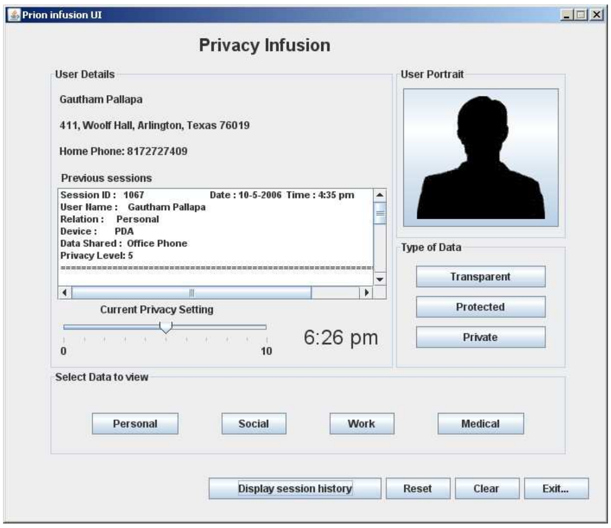
Figure 8 shows a portion of the table in the privon cache. Table I captures the advantage of using privons in a ubiquitous environment. We have considered a set of events in which a user requests for information. In the absence of privons, this information is transparent to users in the environment. On deploying privons, the nature of data changes, depending on the privacy level set by the user.

To illustrate data classification in our scheme, let us consider an entry for a data element in Figure 8. Let us consider $\hat{w}_1 = 0.516$ of session 1237. The weight implies that u_1, the data element corresponding to $\hat{w}_1$, is classified as protected data. Let us assume that the user decreases the privacy level of the session by $\delta = 0.5$. Then, $\text{Dec}_{0.5}(\hat{w}_1) = 0.718$, changing u_1 into transparent data. On the other hand, if the privacy level is increased by $\delta = 0.7$, $\text{Inc}_{0.7}(\hat{w}_1) = 0.3886$, making u_1 private.



Session ID	Privacy Coefficient (α)	Weights									
		w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_{10}
1067	2	0.632	0	0	0.411	0	0	0	0.271	0.239	0
1231	3	0.478	0.361	0	0	0.248	0.253	0	0	0.181	0.134
1237	3	0.516	0	0	0.317	0.299	0	0	0.174	0.193	0
1462	2	0	0.615	0.597	0	0	0.432	0.486	0	0	0.112
1753	5	0.454	0	0.398	0.283	0.272	0.209	0	0.127	0	0.109

Fig. 8. Entries in the privon cache



Prion infusion UI

Privacy Infusion

User Details

Gautham Pallapa

411, Woolf Hall, Arlington, Texas 76019

Home Phone: 8172727409

Previous sessions

Session ID : 1067 Date : 10-5-2006 Time : 4:35 pm
User Name : Gautham Pallapa
Relation : Personal
Device : PDA
Data Shared : Office Phone
Privacy Level: 5

Current Privacy Setting: 0 – 10

6:26 pm

Fig. 9. User Interface for Privacy Infusion

We have simulated the scenario discussed in Section II using $Java^{TM}$ technology for desktop machines, and $J2ME^{TM}$ [34] for mobile devices. The emulators provided by the Sun Java Toolkit for CDC [35] were used to develop Precision for PDAs. The frontend for Privacy Infusion using Precision is shown in Figure 9. A history of the user sessions can be viewed along with the data shared. The session history lists the time, date, requestor, and privacy level of any prior request. A slider is provided, by which the user can set the privacy level for the entire session. The user can also view the data currently designated as *transparent, protected,* and *private* and make modifications if necessary. Additionally, users can view the stored data based on social, personal, and professional contexts. We are currently working on a light-weight version of Precision for smart phones and developing APIs to port our privacy infusion scheme onto these devices.

V. CONCLUSION

Interpreting privacy is more of a fluid and malleable concept rather than a strict set of rules. To process the abstract nature of personal privacy and convert it into tangible issues, a ubiquitous computing framework encompassing privacy infusion is essential.

This paper is a part of the work on Precision, our system for Privacy Enhanced Context-aware Information Infusion in ubiquitous computing environments. In this paper, we have discussed an innovative scheme to infuse privacy into context-aware information through composite entities of data, services, and privacy, called privons. A learning mechanism capable of predicting the user privacy level based on prior interactions with peers is a future extension of Precision. The security issues of the scheme and formulation of privacy policies have to be investigated and incorporated in Precision.

REFERENCES

[1] Mark Weiser, *The future of ubiquitous computing on campus*, Communications of the ACM, vol. 41, No. 1, Page(s):41–42, 1998.

[2] Radenkovic, M., and Lodge, T., *Engaging the public through mass-scale multimedia networks*, IEEE Journal of Multimedia, Volume 13, Issue 3, Page(s):12 – 15, July – Sept. 2006.

[3] Bell, G., *Auspicious computing?*, IEEE Journal of Internet Computing, Volume 8, Issue 2, March-April 2004, Page(s):83 – 85.

[4] Yong Liu, Connelly, K., *SmartContacts: a large scale social context service discovery system*, Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops 2006, Page(s):388 – 392, 2006.

[5] Storz, O., Friday, A., Davies, N., Finney, J., Sas, C., and Sheridan, J., *Public Ubiquitous Computing Systems: Lessons from the e-Campus Display Deployments*, IEEE Journal of Pervasive Computing, Volume 5, Issue 3, Page(s):40 – 47, July – Sept, 2006.

[6] Huang, E.M., Mynatt, E.D., Russell, D.M., and Sue, A.E., *Secrets to success and fatal flaws: the design of large-display groupware*, IEEE Journal of Computer Graphics and Applications, Volume 26, Issue 1, Page(s):37 – 45, Jan – Feb, 2006.

[7] Westin, A., *Privacy and Freedom*, Atheneum, New York, 1967.

[8] Murakami, Y., *Legal issues for realizing ubiquitous information society*, SICE 2004 Annual Conference Volume 2, Page(s):1751 – 1755, 2004.

[9] Khedo, K.K., *Context-Aware Systems for Mobile and Ubiquitous Networks*, Proceedings of the International Conference on Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies (ICNICONSMCL06), Page(s):123 – 128, 2006.

[10] Trope, R.L., *A warranty of cyberworthiness*, IEEE Security & Privacy Magazine, Volume 2, Issue 2, Page(s):73 – 76, Mar-Apr 2004.

[11] Wuest, B., Drogehorn, O., and David, K., *Framework for middleware in ubiquitous computing systems*, IEEE 16th International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC 2005, vol. 4, Page(s):2262 – 2267, 2005.

[12] R. Agrawal, A. Evfimievski, R. Srikant, *Information sharing across private databases*, In Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data, Page(s):86-97, ACM Press, 2003.

[13] *Medical Privacy - National Standards to Protect the Privacy of Personal Health Information*, http://www.hhs.gov/ocr/hipaa/

[14] Beckwith, R., *Designing for ubiquity: the perception of privacy*, IEEE Pervasive Computing, vol.2, no.2 Page(s):40 – 46, April – June 2003

[15] Tentori, M., Favela, J. and Gonzalez, V.: Quality of Privacy (QoP) for the Design of Ubiquitous Healthcare Applications. Journal of Universal Computer Science Vol. 12, No. 3, Page(s):252 – 269, 2006.

[16] Jason I. Hong and James A. Landay, *An architecture for privacy-sensitive ubiquitous computing*, MobiSys '04: Proceedings of the 2nd international conference on Mobile systems, applications, and services, Page(s):177 – 189, 2004.

[17] Jean-Marc Seigneur and Christian Damsgaard Jensen, *Trust enhanced ubiquitous payment without too much privacy loss*, SAC '04: Proceedings of the 2004 ACM symposium on Applied computing, Page(s):1593 – 1599, 2004.

[18] Heiber, Timo and Marron, Pedro Jose, *Exploring the Relationship between Context and Privacy* In Privacy, Security and Trust within the Context of Pervasive Computing, the Kluwer International Series in Engineering and Computer Science, Springer-Verlag, January 2005.

[19] Jason I. Hong, Jennifer D. Ng, Scott Lederer and James A. Landay.: Privacy risk models for designing privacy-sensitive ubiquitous computing systems. Proceedings of the 2004 conference on Designing interactive systems (2004)

[20] Toivonen, S., Lenzini, G. and Uusitalo, I., *Context-aware Trust Evaluation Functions for Dynamic Reconfigurable Systems*, Proceedings of the Models of Trust for the Web workshop (MTW06), vol.190, May 2006.

[21] O-Hoon Choi, Jung-Eun Lim, Hong-Seok Na, and Doo-Kwon Baik, *Modeling of Situation-Middleware for TPO metadata based on Event-Condition-Action Rule*, Fourth International Conference on Software Engineering Research, Management and Applications, Page(s):423 – 427, 2006.

[22] Loureiro, E., Bublitz, F., Barbosa, N., Perkusich, A., Almeida, H., and Ferreira, G., *A Flexible Middleware for Service Provision Over Heterogeneous Pervasive Networks*, Proceedings of the 2006 International Symposium on on World of Wireless, Mobile and Multimedia Networks WOWMOM '06, Page(s):609 – 614, 2006.

[23] Kang-Woo Lee, Eun-Sun Cho, and Hyun Kim, *An ECA rule-based task programming language for ubiquitous environments*, The 8th International Conference of Advanced Communication Technology, ICACT 2006, vol. 1, 2006

[24] Shchzad, A., Hung Quoc Ngo, Lee, S.Y., and Young-Koo Lee, *A comprehensive middleware architecture for context-aware ubiquitous computing systems*, Fourth Annual ACIS International Conference on Computer and Information Science, Page(s):251 – 256, 2005.

[25] Gross, R., Acquisti, A., *Privacy and information revelation in online social networks*, In Proceedings of the ACM CCS Workshop on Privacy in the Electronic Society (WPES 05). (2005)

[26] DeMichiel L. G., *Enterprise Java Beans Specification Version 2.1*, Sun Microsystems Inc. (2002)

[27] Object Management Group, *CORBA Component Model Specification, v4.0*, (April 2006)

[28] Weiser M., *Hot Topics: Ubiquitous Computing*, IEEE Computer, Volume 26, Issue 10 (Oct. 1993) Page(s):71–72.

[29] Frank P. Coyle, *XML, Web Services, and the Data Revolution*, Addison Wesley Professional, March, 2002

[30] Wolfgang Hoschek, *The Web Service Discovery Architecture*, Proceedings of the 2002 ACM/IEEE conference on Supercomputing, (2002)

[31] Cauldwell P., et al., *Professional XML Web Services*, Wrox Press Ltd., 2001.

[32] Smith, L., Roe, C., Knudsen, K.S., *A* $Jini^{TM}$ *lookup service for resource-constrained devices*, Proceedings of IEEE 4th International Workshop on Networked Appliances, Page(s):135 – 144, 2002.

[33] Jiang, X., J. I. Hong, and J. A. Landay, *Approximate Information Flows: Socially-based Modeling of Privacy in Ubiquitous Computing*. In Proceedings of *Ubicomp 2002*, Page(s):176 – 193, 2002.

[34] Sun Microsystems: Java 2 Platform, Micro Edition. http://java.sun.com/j2me/, 2000.

[35] Sun Java Toolkit for CDC. http://java.sun.com/products/cdctoolkit/