Virtual Walls: Protecting Digital Privacy in Pervasive Environments*

Apu Kapadia[1], Tristan Henderson[2], Jeffrey J. Fielding[1], and David Kotz[1]

[1] Department of Computer Science, Dartmouth College, Hanover, NH 03755, USA
[2] School of Computer Science, University of St Andrews, St Andrews, KY16 9SX, UK

Abstract. As pervasive environments become more commonplace, the privacy of users is placed at increased risk. The numerous and diverse sensors in these environments can record users' contextual information, leading to users unwittingly leaving "digital footprints." Users must thus be allowed to control how their digital footprints are reported to third parties. While a significant amount of prior work has focused on location privacy, location is only one type of footprint, and we expect most users to be incapable of specifying fine-grained policies for a multitude of footprints. In this paper we present a policy language based on the metaphor of physical walls, and posit that users will find this abstraction to be an intuitive way to control access to their digital footprints. For example, users understand the privacy implications of meeting in a room enclosed by physical walls. By allowing users to deploy "virtual walls," they can control the privacy of their digital footprints much in the same way they control their privacy in the physical world. We present a policy framework and model for virtual walls with three levels of transparency that correspond to intuitive levels of privacy, and the results of a user study that indicates that our model is easy to understand and use.

1 Introduction

As sensor-rich pervasive environments become more common, users' privacy will be at increased risk [16]. Sensors can record a user's activities and personal information such as heart rate, body temperature, and even conversations. Users may unwittingly leave "digital footprints" (information about users derived from sensors) that can threaten their privacy. These footprints can be disseminated to applications, or stored for later retrieval, giving rise to useful context-aware applications. For example, applications can involve direct queries from other users ("What is Bob doing now?"), triggers ("Alert me when Bob is nearby"), or higher-level actions triggered by notifications ("Create a virtual meeting when Alice and Bob are free in their offices"). Such applications may be useful, but without adequate precautions, digital footprints may be accessed by unwanted parties. While several proposed mechanisms protect location privacy, location is just one kind of digital footprint, and these mechanisms do not directly apply to *all*

* This research program is a part of the Institute for Security Technology Studies and was supported by the Bureau of Justice Assistance under grant 2005-DD-BX-1091. The views and conclusions do not necessarily reflect the views of the United States Department of Justice. To appear in International Conference on Pervasive Computing 2007,

types of footprints. As the number and variety of sensors grows, it will be cumbersome for users to specify fine-grained policies about who can access which footprints.

We address one specific problem: the *confidentiality of digital footprints*, where we define a digital footprint as contextual information derived from raw sensor readings. By confidentiality, we mean that only authorized users should be able to access footprints as defined by the user's *privacy policy*. We believe that the term "digital footprints" is more intuitive to lay users than "context," since digital footprints evoke a sense of a digital trail that a user may leave in the virtual world. We feel that users will be more motivated to protect the privacy of their "digital footprints" rather than their "context."[3]

We propose a policy framework based on the intuitive concept of "virtual walls" that extends the notion of privacy provided by physical walls into the virtual realm. For instance, users are aware of their physical privacy in a closed room — outsiders cannot see or hear them. In a pervasive environment, however, their virtual privacy could be quite the opposite. Digital footprints from a videocamera and a microphone could expose their privacy in the virtual world, where other users *can* see and hear them by accessing their footprints. Figure 1 shows a meeting room where Alice and Bob have physical privacy, but sensors are disseminating personal footprints, such as their images and speech, to unwanted parties. Using virtual walls, users can "bolster" physical walls by specifying intuitive policies that control access to all their personal footprints in a way that is consistent with their notion of physical privacy. Virtual walls also relieve the burden of specifying separate policies for several footprints, which would be cumbersome in sensor-rich environments.

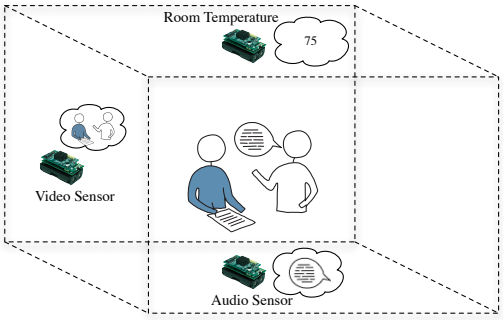

Fig. 1. Alice and Bob are aware of their physical privacy within the meeting room, but sensors are leaking personal footprints to the pervasive environment.

We define three levels of transparency for virtual walls: *transparent*, *translucent*, and *opaque*, each with semantics that match what a user would expect in the physical world. For example, Alice's transparent virtual wall for a room allows users to "see" her personal digital footprints through the wall. An opaque wall restricts the visibility of all footprints within the room, including general footprints such as room temperature or hu-

[3] The term "digital footprints" is not new. Its increasing use in the legal world [2,24] and the popular media [23,27] highlights growing attention to the consequences of digital tracking.

midity. A translucent virtual wall discloses general digital footprints to outsiders (people are present, movement, etc.) while keeping the identities and personal footprints of people hidden. This is similar to physical translucence, through which outsiders cannot identify people, but can see general movement, light, occupancy, and so on.[4] Figure 2 shows a translucent virtual wall that prevents reporting of personal footprints, but keeps general footprints visible. To allow users to better control their privacy, we extend this metaphor by allowing users to create different virtual walls for different queriers. For instance, Alice could create a transparent wall around the cafeteria for her friends, but a translucent wall for her professors. These walls allow her friends to see her personal footprints, but disallow her professors from doing so. We validate the usability of our model through a user study and show that users are indeed able to understand and use our model to express privacy requirements.

This paper makes the following contributions:

A policy abstraction that is easy to understand and use. The metaphor of "virtual walls," validated by a user study, allows lay users to specify their privacy preferences in a way that is consistent with their notion of physical privacy.

Addressing privacy in sensor-rich environments. Users control access to their footprints without having to resort to fine-grained policies for each type of footprint.

A usable interface for specifying policies. We developed a prototype system, and evaluated our graphical user interface (GUI) in a user study. The study validates the ease of use of the GUI and provides valuable insights for further improvement.

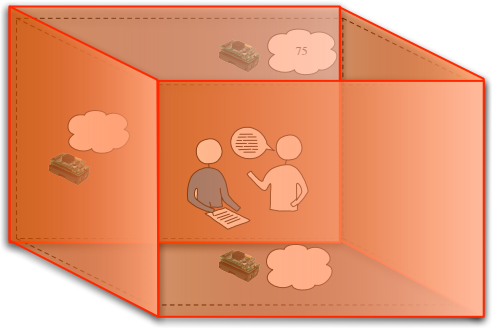

Fig. 2. Alice and Bob deploy a translucent virtual wall to prevent unwanted disclosure of personal footprints.

Next, we outline our system architecture and then describe the virtual walls model in Section 3. Section 4 describes a user study that tested our model and GUI. In Section 5 we discuss some challenges that would arise in a real deployment of virtual walls, and suggest future work. Section 6 discusses related work, and Section 7 concludes.

[4] It is arguably impossible to guarantee complete privacy for users — for example, Alice's location might be hidden while she is in a meeting room, but if Charlie observes Alice enter the room, then her location privacy is compromised. Our goal, however, is to mirror Alice's expectations of privacy in the physical world, where she is already aware of such threats.

2 Architecture

Our system consists of a *context server*, which collects and disseminates digital footprints; clients, who instantiate walls or request footprints; and a sensing infrastructure that extracts footprints from sensor inputs. Users instantiate virtual walls by contacting the context server, which then uses the virtual walls to regulate access to footprints.

We use Solar [4] as the sensing infrastructure in our prototype system, but any such service will do. Solar is an open publish/subscribe framework for processing and distributing contextual events in a pervasive environment. For example, a footprint regarding Bob's current activity can be derived from an accelerometer and body-temperature sensors. Any higher-level inferences (made from raw sensor data) about users' activities generated by the Solar framework are delivered to the context server as footprints containing the activity (e.g., dancing or sleeping), a timestamp, and details about which sensors were used.[5] As we will see, the locations of sensors used to derive the footprint are used to describe where a footprint "originated," and footprint access is controlled in part by their origin. Footprints are treated as soft state, and never recorded to persistent storage. Users query the context server for the most recent footprints, and the footprints are returned if allowed by the virtual walls protecting those footprints. Users create virtual walls by using a GUI, which records the walls in a persistent database at the context server. Figure 3 illustrates the architecture of our system with an example.

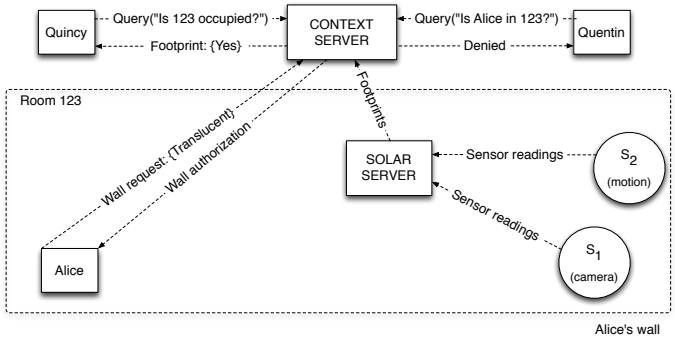

Fig. 3. Alice sets up a translucent virtual wall for Room 123. Raw sensor data are reported to the Solar framework, which generates higher-level footprints such as "Room 123 is occupied." Access to footprints is regulated by the virtual walls for the room. Quincy is granted access to general footprints within Room 123, but Quentin is denied access to Alice's personal footprints.

Our proposed virtual wall system requires various security assumptions:

[5] Solar itself does not make any inferences from raw sensor data, but provides an architecture to do so. We assume that footprints are generated by inferring user activities from raw sensor readings [18,25].

Trusted context server: All users trust the context server in terms of the confidentiality and integrity of their footprints. Only recent footprints are maintained, and only in non-persistent storage, which limits the long-term privacy risk.

Secure location claims: Secure location claims, for instance using a location-limited channel [22], are needed for the creation of opaque virtual walls (Section 3). The location system thus needs to have a level of granularity that is able to identify "places" such as rooms and hallways.

Sensor security: We assume that all sensors are approved by the administrators of the system. We ignore sensors that are deployed by malicious users (e.g., hidden microphones); virtual walls are intended to make pervasive environments more usable by securing *known* devices. We also assume that sensors communicate with the system over a secure channel that provides confidentiality and integrity of sensor data.

3 The Virtual Walls Model

We aim to keep the semantics of virtual walls as intuitive as possible, so that users who specify virtual walls have a clear understanding of who can access their footprints. In keeping with the metaphor of privacy afforded by physical walls, transparent virtual walls allow queriers to access any footprints, even a user's personal information (such as their heart rate or whether they are speaking); opaque walls block access to all footprints originating from within the wall; and translucent walls allow queriers to access only general information such as room temperature and the presence of motion. To add flexibility, users may create walls of varying transparencies for different queriers. For example, Alice may create a transparent virtual wall around her dorm room that applies to her friends and an opaque virtual wall that applies to her professors, thereby allowing finer control over the dissemination of her footprints. In essence, virtual walls are a means for users to specify discretionary privacy policies for their digital footprints.

3.1 The model

To define our model we must describe *places*, *footprints*, *queries* and *virtual walls*.

Places. We refer to physical spaces such as rooms and buildings as *places*. Places have symbolic names, or *labels*, that are readily identifiable by users (e.g., "Room 251, Computer Science Building"), and users may deploy virtual walls around these places. We assume that physical areas are partitioned into non-overlapping or *atomic places* (e.g., a building is partitioned into distinct rooms), and users can define *aggregate places* that map to multiple atomic places. Virtual walls can be defined for any atomic or aggregate place. For example, Alice may specify a virtual wall around the "first floor" place. Our system maps this request to multiple virtual walls around the set of atomic places (rooms and hallways) on the "first floor." We assume a set of predefined places $\mathscr{P}$ and that regions are well-specified, i.e., each has a correct label meaningful to all users. This labeling allows users to specify virtual walls around recognizable places.

Footprints originate from places. Digital footprints derived from sensor data taken from place p are said to *originate* from place p, or places in set $P \subseteq \mathscr{P}$ if they are generated from sensor data in different places or if an individual sensor covers more than one place. For example, the footprint that describes Alice's speech activity, inferred from raw sensor readings taken in Room 251, is said to "originate in Room 251." We categorize footprints into two types: *general*, i.e., those footprints that do not reveal identifiable information about people (such as room temperature), and *personal*, that contain identifiable information (such as Alice's heart rate). We assume that footprints can be classified into one of these two categories, i.e., administrators or programmers can categorize footprints when new footprint generators are defined in the system. Personal footprints contain identifying information about a set of one or more people. For example, "Alice's heart rate" contains information about Alice, and a camera image of a room contains information about the people in that room. We call this set of people the *owners* of the footprint, and the owners of a footprint decide the visibility of that footprint. We assume a predefined set of users $\mathscr{U}$ to which an owner must belong.

Definition 1. *A* **general footprint** *is a tuple $f = \langle d, P, ts, v \rangle$, where descriptor d is a textual description of the footprint such as "Room temperature," $P \subseteq \mathscr{P}$ is the set of places where the footprint originates, ts is the timestamp when the footprint was generated, and v is an object that represents the value of the footprint. A* **personal footprint** *is defined with the same elements plus a set of owners $O \subseteq \mathscr{U}$. Only an owner of a personal footprint can regulate access to that footprint.*

Definition 1 formalizes general and personal footprints. In these definitions we do not concern ourselves with implementation details. Sophisticated implementations would specify d with several attributes (e.g., with an ontology [21]). Our prototype uses a simpler textual representation of d because efficient representation of footprints is orthogonal to the privacy problem that we are addressing. Similarly, the value v of a footprint can be implemented as different data types. Our implementation in Solar provides classes for various types of footprints, which are delivered as "events" in the Solar framework. For simplicity of presentation, we assume that there is only one owner o of a personal footprint, and explain group ownership of personal footprints in Section 3.2.

Footprints are categorized by their descriptor, origin, and owner if any. We refer to these categories as "footprint IDs." The footprint ID for a general footprint $\langle d, P, ts, v \rangle$ is the tuple $\langle d, P \rangle$. Likewise, the footprint ID for a personal footprint is $\langle d, P, o \rangle$. For example the footprint $\langle \textit{room temperature}, \{\textit{Room 241}\}, 16:42:01, 72 \rangle$ has the footprint ID $\langle \textit{room temperature}, \{\textit{Room 241}\} \rangle$. A personal footprint that infers whether Alice is moving may have the ID $\langle \textit{Movement}, \{\textit{Room 241}\}, \textit{Alice} \rangle$. In our implementation, a more recent footprint replaces an older footprint with the same ID and thus the context server only maintains the most recent versions of footprints. Personal footprints are expired from the system after a certain time period so that historical data about users are not recorded. We note that our model can be extended to record historical information.

Query model. We assume that users can query footprints from the context server using any combination of the fields defined for a footprint: place, descriptor, timestamp, owner, and value. For instance, Quentin could ask for a list of all footprints for a place

(such as a conference room), and the context server would return a list of all footprints to which he has access that originate within that place. Similarly, Quentin could query Bob's footprints, such as "Is Bob speaking?" or "Is Bob moving?" Complex descriptors, as used in ontological databases, could enable more general searches. For example, "What is Bob's current activity?" could be mapped onto all footprints that measure Bob's activities such as his motion and speech footprints.

Virtual walls. Virtual walls protect the privacy of users by allowing them to control the visibility of their personal footprints and general footprints in their vicinity. In our implementation, users create virtual walls through a GUI. The context server records walls in a persistent database and uses them to enforce the user's access control policies.

Definition 2. *A* **virtual wall** $w = \langle o, p, t, A \rangle$ *belongs to owner o and protects footprints that originate in place p. We say that virtual wall w is* **around** *place p (e.g., a virtual wall around Room 251). The virtual wall w has* **transparency** $t \in \{\text{transparent}, \text{translucent}, \text{opaque}\}$. *Based on the transparency t, the wall w controls access to footprints originating in p from querying users* $A \subseteq \mathscr{U}$ *where* $\mathscr{U}$ *is the set of users in the system. We call A the* **apply-set***, since the wall w applies to users in A.*

The semantics of transparency are as follows. The transparency t of virtual wall $w = \langle o, p, t, A \rangle$ affects the reporting of digital footprints from place p to the set of users in A. For any personal footprint f_o for owner o and general footprint f_g originating from within the virtual wall w, a querier q

1. *(if t = transparent)* is allowed access to footprint f_o *only if* q is in the apply-set A (not "if," because another wall may block access),
2. *(if t = translucent)* is denied access to footprint f_o if q is in the apply-set A, or
3. *(if t = opaque)* is denied access to f_o and f_g if q is in the apply-set A.

Access to f_g is granted in the absence of an opaque wall. We will discuss the default behavior for access to f_p in the absence of virtual walls further below, but in summary, access to f_p is denied by default, unless the owner (or all the owners) of the footprint allows access with a transparent wall.

The creator of a virtual wall is the owner of that wall. For example the virtual wall $\langle \text{Alice}, \text{Room 256}, \text{translucent}, \{\text{John}, \text{Andy}, \text{Jim}\} \rangle$ is "Alice's translucent virtual wall around Room 256 that applies to her family," or in other words, Alice's family cannot access her personal footprints originating in Room 256. We assume a GUI through which users can define groups such as "Family," which are mapped to multiple users, such as {John, Andy, Jim}, by the system. Our prototype GUI contains this functionality and allows users to define groups, although we did not test this in our user study.

Virtual walls affect access to the *owner's* footprints; Alice's translucent virtual wall will protect *her* personal digital footprints, and not Bob's personal footprints (since Bob is not the owner of that wall). Our model allows different users to set up personal virtual walls around the same place to control access to their own footprints, and each user can create multiple walls for the same place. Section 3.3 discusses interaction between virtual walls created by different users, and how conflicts between walls are resolved.

Notice (in the above semantics) that transparent and translucent walls do not limit access to general footprints that originate within the wall; these are freely available to all users. This policy is based on our assumption that unidentified data do not usually threaten an individual's privacy. For instance, the temperature or occupancy of a publicly-shared place such as a cafeteria is unlikely to affect the privacy of people in that place. For "personal spaces" such as one's home or office, however, general information can be quite revealing — if Bob's office is occupied (a general footprint), then someone is in Bob's office, most probably Bob. In such cases, users can create opaque virtual walls around places where general footprints can result in a breach of privacy. Note that users are already aware of such threats in the physical world (lights on in an office indicate a user's presence), and so an opaque virtual wall is an intuitive countermeasure. Opaque walls may also be useful in public places. For example, if you are known to be the only person who works in a lab after 2am, general footprints from the lab may reveal your location. Since general footprints are not tied to any particular user, the owner of the opaque virtual wall is implicitly claiming ownership of the general footprints within that wall. As this affects other users within the virtual wall, we require unanimous consent from these users. In other words, Bob can freely create opaque walls around a place (even if he is not present in that place), but requires the consent of users present in that place. Section 3.3 discusses this issue in more detail.

In some cases a query may not have virtual walls controlling it, as it is unrealistic to expect that users will specify walls for every possible place and querier. To prevent the accidental release of personal footprints, we block access to personal footprints if there are no virtual walls that apply to a query. In effect, the semantics defined above protect such footprints by a 'default' translucent virtual wall, and users are informed as such. Despite this default behavior, we believe users find it more useful to specify translucent virtual walls explicitly. For instance, Bob may create an opaque wall around a room that applies to all users in the system. If he decides later to make a translucent virtual wall for his family, he could either remove "family" from the apply-set of the opaque wall, and rely on the default behavior, or explicitly create a translucent wall for his family. We believe that the latter is less confusing, and so we allow Bob to explicitly create translucent walls. Furthermore, we are exploring the use of more sophisticated default modes that depend on the type of place or the type of user; e.g., a "free-minded" user could have a default transparent wall, or a "paranoid" user may desire a default opaque policy. Our model also supports "mandatory" system policies, i.e., policies set by administrators that cannot be overridden by users, but we omit discussion for brevity.

3.2 Group ownership

Some personal footprints in the system record data about a *group* of people, for instance, images captured by a camera. We call such footprints with multiple owners, *shared personal footprints*. Access to a shared personal footprint thus needs to be protected by the virtual walls of *all* the "co-owners," and access can be granted only if all these virtual walls permit the access. One could also envision systems in which these co-owners negotiate a "shared virtual wall" for their shared personal footprints. We leave the possibility of negotiation of group policies for future work. For now, we assume

that the system has some way of identifying the group of owners for a shared personal footprint and applies the most restrictive wall of all the owners.

It is possible that unknown users (e.g., users who are not detected or recognized by the system) may be present in a shared personal footprint such as an image. Our system protects the privacy of those users who can be identified in the system; protecting the privacy of unidentifiable users is outside the scope of this work. Posted signs could inform such users that "cameras are present and your images may be broadcast over the network." In other words, users who are not protected by virtual walls implicitly provide informed consent by entering places with cameras. Group ownership of footprints is a powerful concept and Section 5 discusses our thoughts for future work. We now explain how conflicts arise between different virtual walls and how they are resolved.

3.3 Virtual wall interaction

Since multiple virtual walls can be defined for a place, and places themselves may overlap, we need to define the semantics of "conflicting" virtual walls. Two virtual walls *conflict* if there exists a querier for whom two or more different transparencies apply for access to a particular digital footprint. These conflicts can occur 1) between different walls owned by the same user (e.g., if Alice creates both a transparent and a translucent wall for her family), 2) by an opaque wall of one user that contradicts the transparent or translucent virtual walls of other users, or 3) between different walls for different owners for a group-owned personal footprint. For the first case, our framework ensures that conflicts between walls are resolved at creation time, and users are presented with feedback on the conflict. When creating a new wall that conflicts with an old wall, the user is required to choose whether the old or new transparency should be maintained for the affected users in the apply-set, and therefore conflicting walls for the same owner cannot exist within the system for the same place. For footprints with multiple origins, however, the most restrictive wall is applied. For the second case, after all users in the place have agreed to the creation of the opaque walls, the wall is added to the system. If access to a general footprint is restricted by an opaque wall, then access to that footprint is denied, overriding the transparent or translucent walls of the consenting users. For the third case, the most restrictive wall is used. Users are informed of this conflict (users are required to carry a device for creating walls and other interaction with the system) so that they are aware of the restriction placed by the wall of a co-owner of the footprint.

An opaque wall blocks access to the general footprints of all users within a place. Therefore, we require other users present within the place to collectively agree to the presence of an opaque wall. There are several possible approaches for such a negotiation based on who should be given priority in the negotiation. The different negotiation strategies can result in too many, or too few, opaque walls. As such, more work is needed to identify a reasonable strategy. For now, we assume a simple strategy based on unanimous consent. Opaque walls around a place can be removed by a new user entering that place. Entering users are asked whether they agree to the continued existence of the opaque wall. If not, the owner(s) of the opaque wall is given a small time window

(e.g., 5 minutes) after which the opaque wall is removed from the system.[6] Therefore, opaque walls are more effective, and reliably maintained, in places for which the owner has physical control — such as in an office, or in a reserved meeting room with restricted access. An outsider entering a reserved meeting room can be told to leave the room before he or she has the opportunity to disable the opaque wall. We are also exploring the possibility of assigning owners to places (e.g., Alice can be the owner of her personal office), who can then create opaque walls based on authority.

3.4 Limitations of the model

Virtual walls control *all* footprints within the personal and general categories uniformly. Users may want finer-grained control over some footprints (such as location) and specialized mechanisms could be used in conjunction with virtual walls. We leave such a hybrid approach to future work. We have also equated the term "footprints" with context in an effort to make the model simple to understand. Our user study found that users had more difficulty with general footprints than with personal footprints (although the absolute performance numbers were high for both types of footprint). Perhaps another term may be better, but "general footprints" seems workable. Our model does not address multiple queries — Alice may want to restrict the rate of queries for her personal footprints. Lastly, in the physical world, Alice can see observers through a transparent wall and has a sense of her "exposure." In contrast, our model does not provide Alice with information about queriers. Such functionality would need to consider the privacy of queriers as well, and this is an exciting area for future research. It is unclear, however, how far one should push this analogy — at night, observers outside Alice's house can see her through her transparent window, but she may not be able to see them.

4 Evaluation

We built a prototype of our proposed system, including sensors, a context server and a GUI for creating virtual walls. We designed a user study to test the usability of both our model and the GUI; specifically, we tested the ease of understanding of the virtual walls model, the ease of use of the model, and the ease of use of the user interface.

4.1 Study description

We recruited participants using flyers posted around campus, and advertisements on class and departmental e-mail lists, the student-run newspaper, and the popular social networking website "Facebook.com." Participation was *not* restricted to students, and was open to all adults in the community. In total we had 23 participants. We did not record any identifying characteristics such as age or gender, but only asked them whether they had ever taken a programming class, so as to classify the participants by computing experience: 9 participants had never taken a Computer Science (CS) class.

[6] We note that this "5 minute rule" appears arbitrary, and experimentation is required to identify a reasonable time window.

The study comprised a paper booklet and an interactive component that used our GUI on a computer. The booklet contained a four-page introduction to pervasive environments and the virtual-wall system, and three sections of questions as listed below.

1. *Testing the ease of understanding of the virtual walls model.* This booklet section described three different scenarios, where some walls had already been created. In each scenario, participants were asked six questions about whether particular individuals would be able to access different types of data. For example, one scenario included a transparent wall around a dorm room that applied to family members, and asked if a parent could determine if the participant was awake in their dorm room.
2. *Testing the ease of use of the virtual walls model.* This booklet section contained three different scenarios, each with different privacy requirements. Participants were asked to construct walls (with checkboxes in the booklet) that satisfied these requirements. For example, they were asked to set up walls around a lunch room that would allow their friends to query footprints such as what they are eating.
3. *Testing the ease of use of the virtual walls user interface.* Participants followed "wizard" dialogs that introduced our GUI, and then performed tasks to create, modify, or delete walls, where each wall had a maximum of three elements in its apply-set. In each task the participant was told exactly what kind of wall was required — thus these tasks only tested the ability to use the GUI. For instance, participants were asked to set up a transparent wall around a lunch room that applied to friends.

The GUI was implemented using AJAX (Asynchronous JavaScript and XML) so that it could run in a web browser. It used a three-column interface. The first column, shown in Figure 4, contained a list of the participant's current walls, and a dialog box for creating, modifying or deleting walls. This dialog box contained an input box for naming walls, a drop-down box for selecting the room to which the wall applied, and checkboxes to determine transparency or apply-sets. The second interface column contained a map of the area where the walls were being created (in our study, this was a floor of our CS building). Existing walls were displayed on the map using different colors for different transparencies, and clicking on particular rooms would allow a participant to create or modify walls for that room. The third interface column contained a description of the task that the participant needed to perform.

The participants took an average of 28.3 minutes to complete the entire study.

4.2 Study results

We now present more detailed results of our study by breaking down the questions in each section into various categories. For the first two sections, we categorized questions based on the transparency of the wall in the question's scenario, and whether the footprints being accessed were personal or general. For these two sections we define a "Correct Response" as the case where a participant correctly selects or creates the wall to which the question refers. The third section of the study only considered the UI, and so we broke down the questions by the particular UI tasks. For this section a "Correct Response" is the case where a participant successfully performs the task in question; for instance, if a participant was asked to create a wall that applied to "Family" but instead created a wall that applied to "Friends," this would be an incorrect response.

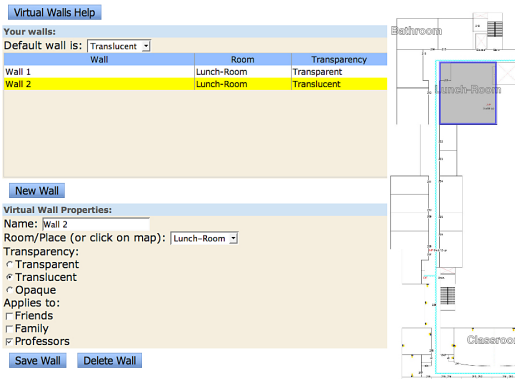

Fig. 4. Portion of the virtual walls user interface

Table 1 shows the correct responses in the three sections of the study. Unsurprisingly, participants who had taken a programming class ("CS participants") performed better than non-CS participants. All of the successful response rates are high and point towards a usable model. In the following we break down the results by section.

Table 1. Overall study responses.

	Correct responses	
Section	*CS participants*	*non-CS participants*
1. Ease of understanding the model	99.4%	90.1%
2. Ease of use of the model	96.3%	90.5%
3. Ease of use of the user interface	97.2%	95.5%

Table 2. Successful responses by topic of question

(a) Section 1, understanding the model

Topic	*Personal Footprints*	*General Footprints*
Transparent	95.7%	91.3%
Translucent	93.5%	88.0%
Opaque	100.0%	95.7%

(b) Section 2, use of the model

Topic	*Correct responses*
Transparent	93.5%
Translucent	94.8%
Opaque	86.96%

Section 1: ease of understanding the model. Table 2(a) shows that participants had more difficulty with the concept of general footprints than personal footprints. Likewise, participants had the most difficulty with translucent walls. This was to be expected — transparent walls allow access to all footprints, and opaque walls restrict access to all footprints, leaving little room for error. Translucent walls, on the other hand, are

Table 3. Time to complete interface tasks.

	Average (Standard deviation) time to complete task (seconds)		
Section	*CS participants*	*non-CS participants*	*non-CS (outlier removed)*
3a. Creating a wall	30.4 (13.7)	31.3 (19.2)	27.7 (11.9)
3b. Modifying a wall	11.6 (5.8)	15.1 (21.3)	10.9 (10.4)
3c. Deleting a wall	11.1 (4.2)	11.3 (9.6)	9.3 (4.7)
3d. Resolving a wall conflict	31.4 (11.2)	37.0 (26.2)	34.6 (23.5)

Table 4. Successful responses for interface tasks.

	Correct responses	
Section	*CS participants*	*non-CS participants*
3a. Creating a wall	94.1%	92.9%
3b. Modifying a wall	100.0%	100.0%
3c. Deleting a wall	100.0%	96.4%
3d. Resolving a wall conflict	100.0%	92.9%

a combination of the two, and showed the highest error rates. In all six categories, however, the success rates were 88% or more, which point toward a usable model.

Section 2: ease of use of model. With regards to use of the model, it was only practical to categorize questions by the transparency of the wall that participants were asked to create (Table 2(b)). While opaque walls were easy to understand when presented to participants (Table 2(a)), three users did not use opaque walls correctly. A better explanation of opaque walls or a few more training examples might help users avoid such mistakes. We thus plan to improve our GUI to provide better feedback, e.g., through a dialog box that says "To block general footprints, you should create an opaque wall." After users become familiar with the model, they can choose to suppress such feedback.

Section 3: ease of use of interface. Table 3 shows the time taken to perform various tasks in the GUI. Resolving conflicts took the longest time, as this requires users to read a dialog box explaining the conflict, and pick the correct option to resolve the conflict. Creating a wall took about the same time, as this requires the largest number of commands: clicking on a room or the "new wall" button, selecting a transparency and an apply-set (restricted to a maximum of three elements), and clicking the "save wall" button. Modifying a wall involved changing just one wall element, and thus the times are similar to those for deleting a wall (which involves just one action to delete). There is a high variance in the responses for the non-CS participants, most of which can be explained by one outlier participant who appeared to have trouble with most of the tasks in the study. We have removed this outlier in the third column of Table 3, but even after doing so there is a high variance in the times taken to resolve conflicts, as there were two participants who took a very long time to do this. Even with these outliers, the longest

task took about half a minute, which we believe makes our current GUI usable (this was reiterated by the participants' positive comments as described below). The next revision of our GUI, however, will concentrate on streamlining the wall-creation and conflict-resolution processes and incorporating users' suggestions for improvement.

Overall, participants were able to use the interface successfully (Table 4). There were two participants that had trouble with creating walls (for example, one user specified "Family" instead of "Professors" in the apply set), but apart from this the other mistakes were infrequent and appeared to be random.

User comments. The study offered participants the opportunity to provide feedback on the system through free-form written comments in the booklet. Many comments implied that participants would be protective of their personal information in a pervasive environment. For instance, one noted "Personally, I don't want people to be able to search the internet for what I am wearing, eating, etc," while another said "I'd refuse to have this kind of software following me." Within the constraints of our system, one participant said that "I think the default wall should be opaque in some cases," while another generalized by saying that "I think many people would choose 'opaque' for the majority of rooms/situations." The study helped us recognize that users may want default opaque walls for personal spaces. Another participant emphasized the need for a secure system: "[I]n the era of identity theft, there is heightened concern about privacy. How easy would it be to get someone's password and reconfigure their Virtual Walls?"

Some comments referenced confusion about the walls concept that was reflected in participants' answers to questions in the study. Two users were confused about locations: one said "I didn't quite udnerstand (*sic*) whether one could query about a specific person without knowing what room they are in" while another was "confused as to location and translucent walls — I understand that translucent would mean they could see people in [the] room but not which people." This indicates that the query model must be explained better to users of the system. Comments regarding the GUI itself were generally positive. Many requested further functionality, such as hotkeys, the ability to define rooms (for simplicity the GUI used in the study only had predetermined rooms), or sounds and additional colors to highlight particular events or interface components.

4.3 Limitations of the study

Our study focused on the usability of the virtual walls *model*. We note that the GUI used in the study was designed to fit in a web browser on a desktop computer. We need to further explore scaling this GUI to smaller displays, such as on a mobile phone or PDA. We believe that our use of standard AJAX technologies should, however, facilitate the porting of our interface to such devices. It would also be useful to evaluate our model against other metaphors for usable privacy policies, but we leave this to future work.

5 Discussion and future work

Our user study affirms that the virtual walls model is easy to understand, and users can effectively translate privacy preferences into policies with virtual walls. For a real-world deployment, we anticipate the following challenges and opportunities for future work.

Creating walls: It may be cumbersome for users to constantly think about deploying virtual walls for every place they visit. A usable system will need to support higher level rules (in addition to default virtual walls) so that users can create virtual walls for not only a particular room, but a set of rooms based on certain conditions. For example, "Transparent wall around all work-related rooms during work hours," or "Translucent wall around current room if I am with my spouse."

Group ownership: A group of users may want to create a shared virtual wall to control footprints related to the group. For example, a group in a room may negotiate a shared translucent wall to restrict access to all personal footprints belonging to the group. Enabling groups to set up walls and negotiate their transparency and apply-set would require an extension to our model and a usable mechanism for such negotiation.

User disruption: Opaque walls, and possibly group walls in the future, require input from and feedback to users. In places with many users entering and leaving, users in those places may be disrupted by several messages. To reduce user input, users' responses could be automated by using stored preferences.

Data perturbation: It may be desirable to perturb footprints (e.g., changing granularity [9], darkening images [14] or adding "noise" [20]) when using translucent walls. This approach would be closer to the physical metaphor of translucence.

Mobile places: One can envision mobile places, such as a bus enriched with sensors. For instance, the bus may be parked inside a building and so be affected by virtual walls around that building. We would like to explore the semantics of virtual walls for mobile places, and how they interact with static places.

Deception: User studies in location privacy [15] have identified the need for deception, where users can lie about their location. Given the broad range of digital footprints that our framework is targeting, deception may be a challenging task.

6 Related work

Several context-dissemination systems in pervasive computing support access-control mechanisms for protecting sensitive context information. Dey [7] built an experimental mechanism to control access to context in the Context Toolkit. The developer of a widget object can specify the "owner" of the information being sensed by implementing a function that computes the owner of an event. As mentioned earlier, we assume some such mechanism to infer the owner of a footprint.

While sensor-derived context information is still an active area of research [18,25], many applications based on context "sensed" from the user's computer, and systems to deal with the privacy of context information, are being studied. IBM's Grapevine service provides a user's context information to other users. A user's context is computed by monitoring her activities on her computer, and other users in the system can check to see her activity before initiating communication. Christensen et al. [5] report that while other users found it useful to query a user's context in certain situations, revealing context was a sensitive issue for most users and they ended up blocking context to all queriers. Fine-grained access control mechanisms were rarely used. It appears that in the absence of a usable policy language, users will be burdened by fine-grained policies, and end up being loathe to part with their (sensitive) context information. A policy

language that balances ease of use and the granularity of access control is therefore needed, and our virtual walls system is an attempt to meet this balance.

A context-privacy system similar to ours is the "Digital Territory" project [3]. This project proposes "bubbles," which are "a temporary defined space that can be used to limit the information coming into and leaving the bubble in the digital domain." Bubbles can be shared between individuals and groups, and the flow of information in and out of the bubble can be adjusted. The bubble is similar to our virtual wall, although instead of translucency, bubbles offer a larger number of policies, which we believe will be unmanageable for complex environments. As far as we know, their system has yet to be implemented, and our translucency concept may prove useful for making the system usable. The pawS [17] system is also similar to ours — it sends privacy beacons to users as they enter pervasive environments to inform them of privacy policies. pawS concentrates on using machine-readable policies, however, while our focus is on policies that are easy for users to understand. Wickramasuriya et al. [26] examine context privacy in media spaces. They use RFID tags for localization, and then start monitoring users (through video sensors) only when policy violations occur (such as movement into a specific area). We anticipate that virtual walls will be used in environments where sensors are not just used for policy enforcement, and that users may opt for continuous monitoring due to the perceived benefits of the resulting context-aware applications.

Location is a primary piece of information for context-aware computing, and so several systems provide access-control mechanisms for location. Geopriv [6] defines a framework for securely disseminating location data by distributing location objects that are coupled with privacy rules. Geopriv, does not, however, address how privacy rules are defined. Hengartner and Steenkiste [12] support two types of authorization policies for location information: user policies specify who is allowed to access their location information, while room policies state who has the privilege to identify people in a particular room. The room policy is similar to virtual walls in that it is associated with a certain geographical area. Unlike our system, the owner of a room (not the users in that room) determines the room policy of that room. Other systems such as Confab [13] and LocServ [19] allow users to specify fine-grained policies to control their location information. For example, LocServ's authorization language expresses constraints such as time, location, and quality of service (i.e., the granularity and anonymity of location information). As mentioned earlier, we provide a policy language for expressing privacy policies about all kinds of footprints, of which location is only one example.

We believe that virtual walls will make privacy in pervasive environments more usable, in particular where large numbers of sensors and users are involved. The idea of using room or wall-like metaphors has been applied in non-security scenarios, for instance by Henderson and Card [11]. Similarly, the idea of simplifying privacy policies into easily-understandable levels has been explored elsewhere, for example, Hawkey and Inkpen's "privacy gradients" [10]. Usability in sensor-network scenarios, however, has been little studied. Barkhuus and Dey examine location-tracking services and find that users are more concerned about privacy if their location is being tracked, rather than if the device is simply aware of its own position [1]. Iachello et al. [15] develop and trial a location-aware application, and produce a set of guidelines for application designers, which we discussed in Section 5. Finally, Elliot et al. [8] conducted interviews to study

how households handle communication information at home, and find that people do attach ownership to information according to the ownership of physical spaces.

7 Conclusion

In this paper we outline the need for an intuitive policy abstraction to address the privacy of users' digital footprints in sensor-rich pervasive environments. We believe that most users will be incapable of specifying complex privacy policies for information sensed about them, and so an abstraction for this purpose must be easy to understand and use. To meet these goals, we proposed and evaluated a policy framework that extends the metaphor of physical walls to pervasive environments. Virtual walls control the spread of personal and general contextual data, or "footprints," and offer privacy analogous to that afforded by physical walls. We formalized the semantics of access control using virtual walls, and evaluated our model through a user study. Our results indicate that the model is easy to understand (users can correctly identify the behavior of virtual walls), and easy to use (users can translate privacy requirements into an appropriate set of virtual walls). Moreover, comments from study participants indicate that privacy in sensor-rich environments is an important problem that might affect the deployment of such environments. Based on our results, we believe that virtual walls are a promising metaphor for specifying usable privacy policies in pervasive environments.

Acknowledgments

We would like to thank Kazuhiro Minami, Harald Vogt, the Dartmouth CMC Lab and the anonymous reviewers for their helpful comments, and Ron Peterson for suggesting the original concept. We also thank Alex Iliev, Chris Masone, Sara Sinclair, and Phoebe Wolfskill for pre-testing our user study.

References

1. L. Barkhuus and A. Dey. Location-based services for mobile telephony: a study of users' privacy concerns. In *Proceedings of the 9th IFIP TC13 International Conference on Human-Computer interaction (INTERACT 2003)*, Zürich, Switzerland, Sept. 2003.
2. M. H. Barrera and J. M. Okai. Digital correspondence: Recreating privacy paradigms. *International Journal of Communications Law and Policy*, 1(3), Summer 1999.
3. L. Beslay and H. Hakala. Digital territory: Bubbles, 2005. Draft publication. http://cybersecurity.jrc.es/docs/DigitalTerritoryBubbles.pdf.
4. G. Chen, M. Li, and D. Kotz. Design and implementation of a large-scale context fusion network. In *Proceedings of Mobiquitous 2004*, pp 246–255, Boston, MA, USA, Aug. 2004.
5. J. Christensen, J. Sussman, S. Levy, W. E. Bennett, T. V. Wolf, and W. A. Kellogg. Too much information. *ACM Queue*, 4(6):50–57, July-Aug. 2006.
6. J. R. Cuellar, J. B. Morris Jr, D. K. Mulligan, J. Peterson, and J. M. Polk. Geopriv requirements, Feb. 2004. RFC 3693.
7. A. K. Dey. *Providing Architectural Support for Building Context-Aware Applications*. PhD thesis, College of Computing, Georgia Institute of Technology, Dec. 2000.

8. K. Elliot, C. Neustaedter, and S. Greenberg. Time, ownership and awareness: The value of contextual locations in the home. In *Proceedings of UbiComp 2005*, pp 251–268, Sept. 2005.
9. M. Gruteser and D. Grunwald. Anonymous usage of location-based services through spatial and temporal cloaking. In *Proceedings of MobiSys 2003*, pp 31–42, San Francisco, CA, USA, May 2003.
10. K. Hawkey and K. M. Inkpen. Privacy gradients: exploring ways to manage incidental information during co-located collaboration. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems*, pp 1431–1434, Portland, OR, USA, Apr. 2005.
11. D. A. Henderson, Jr. and S. K. Card. Rooms: the use of multiple virtual workspaces to reduce space contention in a window-based graphical user interface. *ACM Transactions on Graphics*, 5(3):211–243, July 1986.
12. U. Hengartner and P. Steenkiste. Protecting access to people location information. In *Proceedings of the First International Conference on Security in Pervasive Computing*, pp 25–38, Boppard, Germany, Mar. 2003.
13. J. I. Hong and J. A. Landay. An architecture for privacy-sensitive ubiquitous computing. In *Proceedings of MobiSys 2004*, pp 177–189, Boston, MA, USA, June 2004.
14. S. E. Hudson and I. Smith. Techniques for addressing fundamental privacy and disruption tradeoffs in awareness support systems. In *Proceedings of the 6th ACM Conference on Computer Supported Cooperative Work*, pp 248–257, Boston, MA, USA, Nov. 1996.
15. G. Iachello, I. Smith, S. Consolvo, M. Chen, and G. D. Abowd. Developing privacy guidelines for social location disclosure applications and services. In *Proceedings of the 2005 Symposium on Usable Privacy and Security*, Pittsburgh, PA, USA, July 2005.
16. M. Langheinrich. Privacy by design - principles of privacy-aware ubiquitous systems. In *Proceedings of UbiComp 2001*, pp 273–291, Atlanta, GA, USA, Sept. 2001.
17. M. Langheinrich. A privacy awareness system for ubiquitous computing environments. In *Proceedings of UbiComp 2002*, pp 237–245, Göteborg, Sweden, Sept. 2002.
18. J. Lester, T. Choudhury, and G. Borriello. A practical approach to recognizing physical activities. In *Proceedings of Pervasive 2006*, pp 1–16, Dublin, Ireland, May 2006.
19. G. Myles, A. Friday, and N. Davies. Preserving privacy in environments with location-based applications. *IEEE Pervasive Computing*, 2(1):56–64, Jan.-Mar. 2003.
20. B. A. Price, K. Adam, and B. Nuseibeh. Keeping ubiquitous computing to yourself: A practical model for user control of privacy. *International Journal of Human-Computer Studies*, 63(1-2):228–253, July 2005.
21. A. Ranganathan, J. Al-Muhtadi, and R. H. Campbell. Reasoning about uncertain contexts in pervasive computing environments. *IEEE Pervasive Computing*, 3(2):62–70, Apr.-June 2004.
22. N. Sastry, U. Shankar, and D. Wagner. Secure verification of location claims. In *Proceedings of the ACM Workshop on Wireless Security*, pp 1–10, San Diego, CA, USA, Sept. 2003.
23. B. Schneier. Your vanishing privacy. *The Star Tribune*, page 1AA, Mar. 05, 2006.
24. P. Sommer. Digital footprints: Assessing computer evidence. *Criminal Law Review*, pp 61–78, Dec. 1998.
25. E. M. Tapia, T. Choudhury, and M. Philipose. Building reliable activity models using hierarchical shrinkage and mined ontology. In *Proceedings of Pervasive 2006*, pp 17–32, Dublin, Ireland, May 2006.
26. J. Wickramasuriya, M. Datt, S. Mehrotra, and N. Venkatasubramanian. Privacy protecting data collection in media spaces. In *Proceedings of the 12th Annual ACM International Conference on Multimedia*, pp 48–55, New York, NY, USA, Oct. 2004.
27. E. Wieffering. Protecting your digital footprints. *The Star Tribune*, page 1D, Nov. 07, 1999.