Privacy Risk Models for Designing Privacy-Sensitive Ubiquitous Computing Systems

Jason I. Hong, Jennifer D. Ng, Scott Lederer
Group for User Interface Research
Computer Science Division
University of California, Berkeley
Berkeley, CA USA
jasonh@cs.berkeley.edu

James A. Landay
DUB Group
Computer Science and Engineering
University of Washington
Seattle, WA USA
landay@cs.washington.edu

ABSTRACT

Privacy is a difficult design issue that is becoming increasingly important as we push into ubiquitous computing environments. While there is a fair amount of theoretical work on designing for privacy, there are few practical methods for helping designers create applications that provide end-users with a *reasonable* level of privacy protection that is commensurate with the domain, with the community of users, and with the risks and benefits to all stakeholders in the intended system. Towards this end, we propose *privacy risk models* as a general method for refining privacy from an abstract concept into concrete issues for specific applications and prioritizing those issues. In this paper, we introduce a privacy risk model we have developed specifically for ubiquitous computing, and outline two case studies describing our use of this privacy risk model in the design of two ubiquitous computing applications.

Categories and Subject Descriptors

H.5.2 [**Information Interfaces and Presentation**]: User Interfaces—Theory and methods, Style guides, Evaluation/methodology; K.4.1 [**Public Policy Issues**] – Privacy

General Terms

Design, Human Factors

Keywords

Privacy, Privacy Risk Model, Ubiquitous Computing, Method

1. INTRODUCTION

Privacy has always been a contentious issue for ubiquitous computing. On the one hand, the convergence and increasing widespread deployment of sensors, wireless networking, and devices of all form factors are providing tremendous opportunities for interaction design, allowing us to create systems that can improve safety, efficiency, and convenience. On the other hand, there are numerous interviews (e.g. [7, 19]), essays (e.g. [11, 38, 40]), books (e.g. [10, 15]), and instances of negative media coverage (e.g. [37, 42]) that indicate a general unease over the potential for abuse, fear over a potential lack of control, and desire for privacy-sensitive ubicomp systems. These concerns suggest that privacy may be the greatest barrier to the long-term success of ubiquitous computing.

This barrier persists, in part, because it is difficult to design privacy-sensitive ubiquitous computing systems. Discussions about privacy often generate a great deal of heat but little light. There are two primary reasons for this. The first is the wide range of issues that fall under the rubric of "privacy", including concepts as wide-ranging and disparate as Big Brother governments watching every move you make, overprotective parents keeping close tabs on their children, overzealous telemarketers, and protection of one's genetic information. The second reason is that we each perceive privacy differently. As Westin notes, "no definition [of privacy]… is possible, because [those] issues are fundamentally matters of values, interests and power" [4]. As a result, it is difficult to sort out and conduct reasoned debates over the practical issues, and then to design systems that address them effectively.

Our position is that a systematic method is needed to help designers identify, understand, and prioritize privacy risks for specific applications. Here, the goal is not perfect privacy (if there even is such a thing), but rather a practical method to help designers create applications that provide end-users with a *reasonable* level of privacy protection that is commensurate with the domain, the community of users, and the risks and benefits to all stakeholders in the intended system.

Towards this end, we propose *privacy risk models* as a general method for doing this. Herein we focus on *personal privacy*, the processes by which individuals selectively disclose personal information–such as email address, shopping history, or location– to organizations and to other people. We also introduce a specific privacy risk model for personal privacy in ubiquitous computing.

Our privacy risk model consists of two parts. The first part is a *privacy risk analysis* which poses a series of questions to help designers think about the social and organizational context in which an application will be used, the technology used to implement that application, and control and feedback mechanisms that end-users will use. The second part looks at *privacy risk management*, and is a cost-benefit analysis intended to help designers prioritize privacy risks and develop architectures, interaction techniques, and strategies for managing those risks. This privacy risk model is intended to be used in conjunction with other methods, such as interviews and lo-fi prototypes.

This privacy risk model came about from an analysis of previous work, an examination of emerging ubicomp applications in use (most notably AT&T Wireless Find Friends [6]), as well as from our own experiences in developing privacy-sensitive systems. We noticed that there were many common patterns of issues with respect to privacy, and so we compiled them into a format more amenable for design teams.

The rest of this paper is organized as follows. First, we place privacy risk models in the context of related work. Then, we describe our privacy risk model in detail. We wrap up with two case studies describing our use of the privacy risk model in developing two ubicomp applications, a location-enhanced instant messenger and an emergency response service.

2. RELATED WORK

There has been some previous analytical and prescriptive work on privacy-sensitive systems. Bellotti and Sellen argue the importance of feedback and control for maintaining privacy in multimedia ubicomp environments [9]. Palen and Dourish argue that privacy is not simply a problem of access control, but is rather an ongoing and organic process of negotiating boundaries of disclosure, identity, and time. They also suggest *genres of disclosure* as a sort of design pattern approach to support the development of privacy-sensitive applications [32]. Langheinrich looked at how the fair information practices can be adapted for ubicomp scenarios, providing many examples of how these practices might influence the design of such applications [22]. Jiang et al proposed a systems design space for minimizing the information asymmetry between users and observers [21]. Lederer et al provide a useful deconstruction of the privacy space, looking at system properties, actor relations, and information types [24]. The privacy risk model we propose is inspired by the theoretical work above, but is focused more on providing a practical method that designers can use to concretely conceptualize and mitigate privacy risks faced by end-users in specific domains. In a related paper, we offer a set of pitfalls in designing user interfaces for privacy [23].

A commonly cited resource in the privacy canon is the set of fair information practices. These guidelines help large organizations, such as corporations and governments, manage people's personal information in a responsible manner [41]. They include concepts such as notice, choice, security, and recourse. While extremely influential on the field of information privacy and on this work as well, the fair information practices are intended more for large organizations and do not translate well for interpersonal relationships, such as between friends and family. Furthermore, the fair information practices provide high-level requirements, rather than delving into specific privacy risks. The privacy risk model we propose is complementary to the fair information practices, in that it can help designers examine specific privacy risks for specific domains and end-users. It can also aid designers in determining what kinds of security and recourse mechanisms are needed, helping to translate these high-level requirements into more concrete and detailed goals.

From an interaction design perspective, creating a privacy risk model is similar in spirit to performing a task analysis (see for example [18]). A task analysis involves asking a systematic series of questions about the end-users, their desired tasks, their current tools, and their social and organizational context. The privacy risk

model we propose falls along these lines, but focuses on specific privacy-related factors, rather than on the task as a whole.

Privacy risk models were inspired by the idea of security threat models in the field of computer security. Felten, a well-known security researcher, describes the importance of security threat models as follows:

[T]he first rule of security analysis is this: understand your threat model. Experience teaches that if you don't have a clear threat model - a clear idea of what you are trying to prevent and what technical capabilities your adversaries have - then you won't be able to think analytically about how to proceed. The threat model is the starting point of any security analysis. [14]

Our goal with the privacy risk model is to do the same, focusing on privacy for individuals rather than on security for the systems that those individuals use. Here, it is important to draw a distinction between *security* and *privacy*. Saltzer and Schroder [34] describe security as the "mechanisms and techniques that control who may use or modify the computer or the information stored in it", and privacy as "the ability of an individual (or organization) to decide whether, when, and to whom personal (or organizational) information is released."

Security and privacy are clearly related; however, while a basic level of security is necessary for helping people manage their personal privacy, it is by no means sufficient. Furthermore, the security mindset is often very different from what is needed in developing privacy-sensitive applications. In security, one is often defending against adversaries that are actively attacking and threatening one's systems and resources. However, Orwell and media headlines notwithstanding, this is not always the case with privacy. For example, one could imagine sharing one's location information with friends to facilitate micro-coordination of arrivals at a meeting place, or sharing simple notions of activity to convey a sense of presence to co-workers and friends. It is important to note here that in these cases, the parties that are receiving such information already know one's identity, are not adversaries in the traditional sense, and that the privacy risks may be as simple as wanting to avoid undesired social obligations or potentially embarrassing situations.[1]

The point is that, rather than being a single monolithic concept, privacy is a heterogeneous, fluid, and malleable notion with a range of needs and trust levels. The goal of a privacy risk model is to help elucidate those needs and trust levels, refining privacy from abstract principles into concrete issues that can be acted upon in specific domains for specific applications.

As a final note, privacy risk models tend to look at privacy from the perspective of individual end-users and their relationships, rather than that of large communities. In some cases, it may be of greater benefit to the overall community not to have some forms of privacy, for example making it mandatory to display license plates on cars. Etzioni [12] calls this the communitarian view on privacy, and discusses the balance between privacy for individuals and benefit for communities with respect to such topics as mandatory HIV testing, sex offender laws, and medical records. This topic, however, is beyond the scope of this paper.

---

[1] These differences are also why we termed our method a *privacy risk model* rather than the *privacy threat model*.

3. A PRIVACY RISK MODEL FOR UBIQUITOUS COMPUTING

In this section, we describe a privacy risk model that we have developed for ubiquitous computing, though aspects of it will apply to networked applications in general. Our privacy risk model is comprised of two parts. The first part is a *privacy risk analysis* that poses a series of questions to help designers refine their understanding of the problem space. The second part looks at *privacy risk management*, which deals with categorizing, prioritizing, and developing interaction techniques, architectures, and strategies for managing potential privacy risks.

3.1 Privacy Risk Analysis

The first part of our privacy risk model is a set of questions intended to help design teams think through specific privacy issues for ubiquitous computing applications. The output from this analysis should be an unordered list of potential privacy risks.

We have organized these questions into two groups, looking at the social and organizational context in which an application is embedded and the technology used in implementing that application. These questions include:

Social and Organizational Context
- Who are the users of the system? Who are the *data sharers*, the people sharing personal information? Who are the *data observers*, the people that see that personal information?[2]
- What kinds of personal information are shared? Under what circumstances?
- What is the value proposition for sharing personal information?
- What are the relationships between data sharers and data observers? What is the relevant level, nature, and symmetry of trust? What incentives do data observers have to protect data sharers' personal information (or not, as the case may be)?
- Is there the potential for malicious data observers (e.g., spammers and stalkers)? What kinds of personal information are they interested in?
- Are there other stakeholders or third parties that might be directly or indirectly impacted by the system?

Technology
- How is personal information collected? Who has control over the computers and sensors used to collect information?
- How is personal information shared? Is it opt-in or is it opt-out (or do data sharers even have a choice at all)? Do data sharers push personal information to data observers? Or do data observers pull personal information from data sharers?
- How much information is shared? Is it discrete and one-time? Is it continuous?
- What is the quality of the information shared? With respect to space, is the data at the room, building, street, or neighborhood level? With respect to time, is it real-time, or is it several hours or even days old? With respect to identity, is it a specific person, a pseudonym, or anonymous?
- How long is personal data retained? Where is it stored? Who has access to it?

---

[2] In existing privacy literature, the terms data subjects (or data owners), data collectors, and data users have been used. We use *data sharers* and *data observers* for their less sinister and less adversarial connotations.

There are five things to note about using the questions described above. First, they should be used to describe the average case first, that is how the application is expected to be normally used. Afterwards, they should be used to describe special cases. The reason for this is that there will always be an endless number of exception cases, but (by definition) average cases are more likely to occur and so it makes sense to make sure that they are addressed first.

Second, these questions can be asked in any order. Third, these questions are not meant to be mutually exclusive. Many of them cover the same issue but from different perspectives.

Fourth, this privacy risk model is neither consistent nor complete. By the former, we mean that different project teams will not always arrive at the same answer. Each team will have their own unique biases, perspectives, and insights. By the latter, we mean that this is only a starting set of questions. Designers may find it useful to add or remove questions depending on the community of users and the intended domain.

Fifth, our privacy risk model addresses relatively few security issues. It should be used in conjunction with a security threat model to ensure that the desired privacy needs are properly implemented and secured.

Below, we examine each of these questions in depth, describing what kinds of information the question is looking for, why it is important, and some examples.

3.1.1 Social and Organizational Context

Who are the users of the system?

The first step in our privacy risk model is to identify potential users. This is an important step because each community of users has different attitudes towards privacy, risk, and sharing of personal information. For example, a system intended for friends to share location information would have different privacy needs than a similar system intended for co-workers, or a system for real-time monitoring of one's health, or a system that makes personal information publicly available on the web.

It is also important to identify the intended data observers in a system, since this is a significant factor in whether or not a data sharer is willing to share personal information, as indicated by Adams' work in privacy in multimedia spaces [3] and Lederer's work in managing end-user privacy in ubiquitous computing environments [25]. Nodder also suggests that data sharers have a range of people across which they are willing to share different types and degrees of personal information [30].

Since it is extremely difficult to enumerate every possible usage scenario, the focus here should be on *likely users* that will be affected by the system in the average and most likely cases. This question should also focus on non-malicious data observers, that is people who may intrude on one's privacy but do not necessarily wish harm to the data sharer (such as an overprotective mother).

What kinds of personal information are shared?

This question looks at what is shared and how it is shared. Enumerating the kinds of information shared and the circumstances in which it is shared helps identify potential privacy risks and suggests ways in which they can be managed.

For example, anonymity is frequently touted as a panacea for protecting privacy. However, most friends and family already know a great deal about an individual's name, address, phone

number, hobbies, and preferences. This last point also underscores another issue, which is to think about what personal information is already known in a relationship. Since family members already know one's name and address, for example, it does not always make sense for a family-oriented application to provide strong protection over those pieces of information[3].

One way of approaching this question is to look at the technology being used. For example, with active badges [20], the kinds of information being shared would be location, and to a rougher approximation, activity. Marx presents an alternative analysis, listing seven types of identity knowledge which we use to characterize people [28]. We can know:

- A person's name, which could be a legal name, or first or last name only;
- A person's address, which could be a mailing address, email address, homepage, blog, or instant messenger address;
- A unique identifier, which could be a social security number or bank account number;
- Names or pseudonyms that cannot be easily traced, for example a disposable identifier used for anonymous HIV testing;
- A person's appearance or behavior, for example web browsing habits, fashion style, or writing style;
- A person's social categorization, including "gender, ethnicity, religion, age, education, region, sexual orientation, linguistic patterns, organizational memberships and classifications, health status, employment, leisure activities… credit risk, IQ, SAT scores, life style categorization for mass marketing" [28]; and
- A person's relationship with others, who they are in love with, who they like, who they dislike, what services they use.

Nodder suggests that people have different sensitivity to different kinds of information as well [30]. For example, domestic, romantic, and financial information, as well as social security numbers, are things that people may be less comfortable sharing, compared to marketing information, good and bad experiences, jokes, stories, and opinions.

There are three notes here. First, it is useful here to consider how ubicomp changes what can and cannot be identified. By capturing physical world information in real-time, ubiquitous computing greatly allows knowledge or inference of where a person is, what that person is doing, and other people currently around.

Second, it is useful to note what information is explicitly versus implicitly shared. Using the active badge example, a data sharer is explicitly sharing their location, but implicitly sharing some notion of activity. Knowing that a person is in the office suggests that they are working (or at least attempting to appear so).

Third, it is also useful to note of how often the data changes. For example, a person's name and birthday are fairly static in that they normally do not change. A person's preferences and habits are semi-static in that they change relatively slowly. However, a person's location and activity is dynamic, in that it can change quite often. This is an important distinction to make because some privacy protection techniques only work well on certain kinds of data. For example, an individual cannot revoke access to one's

name or birthday information and expect any meaningful effect, but could do so with location or activity information.

What is the value proposition for sharing?
This question looks at reasons data sharers have to share personal information. Without a strong value proposition, data sharers may feel that they have no compelling reason to share information, as it exposes them to risk without any benefit. In many ways, this can be considered a variation of Grudin's law [16], which informally states that when those who benefit are not those who do the work, then the technology is likely to fail or be subverted. The privacy corollary is that when those who share personal information do not benefit, then the technology is likely to fail.

One example of this can be seen with nurse locator systems. Some hospitals have their nurses wear active badges, allowing the hospital to track the location of nurses for efficiency and accountability purposes. What is interesting is that comments on a nurse message board [5] about such systems can be divided into two groups. The first group, forming a majority of the comments, is skeptical and distrusting of such locator systems and in some cases even rejected those systems, making arguments such as "I think this is disrespectful, demeaning and degrading" and "I guess my question is how does this help the NURSE?"

The second group of nurses *was* initially skeptical, but was won over because management did not abuse the system and because they eventually saw the value of such a system. One nurse wrote, "I admit, when we first started using it we all hated it for some of the same reasons cited above [in the message board] but I do think it is a timesaver! It is very frustrating when someone floats to our unit and doesn't have a tracker…can't find them for [doctor] calls, [patient] needs etc." Another nurse echoed this sentiment, writing, "At first, we hated it for various reasons, but mostly we felt we couldn't take a bathroom break without someone knowing where we were…[but now] requests for medications go right to the nurse and bedpans etc go to the techs first. If they are tied up, then we get a reminder page and can take care of the pts needs. I just love [the locator system]."

Thinking about privacy from the perspective of the value proposition also helps to explain many of the recent protests against the proposed deployment of RFID systems in the United States and in England (see for example [8]). From a store's perspective, RFIDs benefited them because they could use these tags for tracking inventory and maintaining steady supply chains. However, from a customer's perspective, they were becoming data sharers and were being exposed to the risk of surreptitious tracking, without any salient benefit to them at all. It was not surprising that people would have serious privacy concerns here.

What are the relationships between data sharers and data observers?
This question looks at identifying the kinds of relationships between users, which is important in understanding the level of trust, potential risks, incentives for protecting a data sharer's personal information, obligations, and mechanisms for recourse.

For example, a close friend would have a strong motivation to not, say, sell the data sharer's personal information. The kinds of concerns in such a relationship might be as simple as wanting to be alone or as complex as going out with another friend that is not mutually liked. The mechanism for recourse here might be asking the data observer to be more considerate in the future.

---

[3] Of course, there are exceptions here, for example if a husband is abusive and his wife is trying to hide. Depending on the scope and scale of the application, it may be useful to add some ways of hiding.

As another example, if a person is using a paid service, then there is a market and very likely a legal relationship as well. In the large majority of cases, a data sharer probably wants to avoid spam and telemarketers. If the data sharer discovers a privacy violation, then mechanisms for recourse include opting out (if the service provides a web page for doing so), switching to an alternative service, or even suing the service.

We define two levels of relationships here, specific and general. A *specific relationship* defines how the data sharer and data observer knows each other (see Table 1). Some examples include "family" and "friends". A *general relationship* describes the class of the relationship. We use Lessig's framework of Market, Social, and Legal forces to roughly categorize these relationships[4] [26].

Specific Relationship	General Relationship
Family	Social
Friends	Social
Acquaintances	Social (weak)
Strangers	Social (weak), Legal
Charities	Social (weak)
Employer	Market, Legal, Social (weak to moderate)
Co-workers	Social (weak to moderate), Legal
Companies	Market, Legal

Table 1. Some example relationships and their categorization.

Are there malicious data observers?
It is also useful to think about what malicious data observers there might be and what kinds of personal information they are interested in. Unlike the data observers identified previously, these data observers do not have a data sharer's best interests in mind. For example, a stalker or an intrusive journalist would be interested in the location of a specific person. A spammer would be interested in the email address of any person. A mugger would be interested the location of any person that is alone.

It is also helpful here to think about this question from a security perspective rather than a privacy perspective, since a data sharer is unlikely to voluntarily share personal information with such people unless they are tricked, oblivious to risk, or misunderstand how a given application works. For example, how would malicious observers obtain such personal information?

Are there other stakeholders?
This question looks at other stakeholders that might be impacted by a given system, whether purposefully or inadvertently. For example, Place Lab [35] is a project whose goal is to provide an inexpensive and large-scale location positioning system. Place Lab uses the wide deployment of 802.11b WiFi access points for determining one's location. A key observation here is that many developed areas have wireless hotspot coverage so dense that cells overlap. By consulting the Place Lab directories, which will continuously map the unique addresses of each wireless hotspot to physical locations, mobile computers and PDAs equipped with WiFi can determine their location to within a city block.

The advantage to this approach is that it provides a privacy-sensitive way of determining one's location, since hotspots can be detected in a passive manner without revealing any information to any other entities. However, Place Lab introduces new privacy

risks for access point owners as it re-purposes a system originally meant only for wireless communication. Place Lab makes the location of many access points widely available, posing a potential privacy risk to their owners. As such, access point owners are now stakeholders in the system whose privacy must also be considered.

3.1.2 Technology

How is personal information collected?
This question looks at some of the technological mechanisms through which personal information is captured in ubiquitous computing systems. For example, a person's activity could be estimated through motion sensors, video cameras, or monitors installed on the computer. Each of these approaches has different tradeoffs in terms of quality of information collected and privacy.

With respect to location-based systems, there are three general approaches for acquiring location [36]. In the *network-based* approach, infrastructure receivers such as cell towers track cellular handsets or other mobile transmitting units. This approach also includes techniques such as computer vision, where the personal information is initially captured on computers outside of the control of the data sharer. In the *network-assisted* approach, the infrastructure works with clients to calculate location. For example, Qualcomm's Enhanced 911 solution uses handsets to receive raw GPS satellite data that is sent to network processors for calculation. In the *client-based* approach, personal mobile devices autonomously compute their own position, as is the case with a GPS unit.

In general, client-based approaches offer the strongest privacy guarantees, since the personal information starts with data sharers and no information is revealed to any other entities unless data sharers choose to do so. However, using network-based or network-assisted approaches may be sufficient, depending on the community of users and the intended domain.

How is personal information shared?
This question looks at what choices data sharers have when sharing personal information. Here, we assume that data sharers know that some personal information has been captured and that they have a choice in how it is shared. In this case, there are two ways to share information. The first is to *push* information to others, for example sending your location information during an E911 call. The second is to let others *pull* information, for example, a friend requesting your location.

In general, there are fewer privacy risks with respect to push applications since personal information is transferred only when data sharers initiate a transaction. The downside is that it may not always be flexible enough for certain kinds of applications, for example, being notified when a person enters a room.

How much information is shared?
This question looks at the quantity of information that is shared with data observers. At one extreme, data observers can see one-time snapshots of dynamic information. At the other, they can see continuous real-time information.

In general, a greater amount of information is more subject to data mining and inferences, and thus potentially exposes data sharers to greater privacy risks. As suggested by Jiang et al's Principle of Minimum Asymmetry [21] and by the fair information practices [41], a general rule of thumb is that applications should be designed such that only the minimum amount of information that

---

[4] Lessig's framework consists of market, social, legal, and technical forces, but we have dropped technical as a form of relationship here, since technology is a means rather than a form of relationship.

is needed is actually shared. Further, it is important to consider the degree of reciprocity in the disclosure.

What is the quality of the information shared?
This question looks at the quality of information with respect to space, time, and identity. In general, the lower the quality of the information shared, the fewer the privacy risks for the data sharer.

In terms of space, an application could be designed to use a less precise form of location (or let a data sharer choose to share her location at such a level). For example, knowing that a person is at 123 11th Street exposes an individual to different risks (and different opportunities) than knowing only that she is in Chicago.

In terms of time, an application could be designed to restrict the personal information that is being shared based on the temporal granularity ("I was at Lake Tahoe sometime last month" versus "I was at Tahoe July 1 2003") as well as by temporal freshness ("You can have my location information if it is over a week old, but not my current location").

In terms of identity, applications can be designed to use less precise forms of identity. There is a significant difference between a smart room that senses that "a person" versus "a woman" or "alice@blah.com" is inside the room. Marx's work on different forms of identification is especially useful here [28]. However, in general, the kinds of identity that can be used will be dictated by the capabilities and limitations of the underlying technology.

Again, as a general rule of thumb, it is useful to require only the minimum amount of information that is needed, to minimize potential privacy risks to individuals.

How long is information retained?
Limited retention of personal information is an issue explicitly mentioned in the fair information practices [41]. The danger is that retention exposes data sharers to unexpected risks, such as data mining or the use of data that is out of date.

While it is a judgment call as to precisely how long personal information should be retained, the fair information practices provide some guidance. That is, personal information should be retained only for the time necessary to fulfill the initial purpose for which the information was collected.

3.2 Privacy Risk Management

The second part of our privacy risk model looks at privacy risk management, which takes the unordered list of privacy risks from the privacy risk analysis, prioritizes them, and helps design teams identify solutions for helping end-users manage those issues (through technical solutions, social processes, or other means).

This privacy risk management is based on the concept of *reasonable care* in law: "the degree of care that makes sense and that is prudent, enough but not too much" [13]. In a well-known legal ruling, famed jurist Learned Hand proposed a cost-benefit analysis that considers three factors for determining reasonable care with respect to negligence and assigning liability [2]. We adapt Hand's three factors for use in managing privacy as:

- The likelihood L that an unwanted disclosure of personal information occurs
- The damage D that will happen on such a disclosure
- The cost C of adequate privacy protection

We suggest using a qualitative assessment of high, medium, and low, to help gauge each of these factors, though some design teams may find it more useful to use a numerical scale to quantify these values. In general, situations where $C < LD$, that is, where the risk and damage of an unwanted disclosure outweigh the costs of protection, suggest that privacy protections should be implemented. In other words, design teams should provide a *reasonable* level of privacy protection, but not so much that it becomes prohibitive to build and deploy an application or significantly reduces the utility of an application. For example, Agre provides an analysis of CNID (also known as caller ID), and notes that if CNID for callers were off by default (thus protecting some level of privacy for callers), then there would be little value for any callee to have the service [4].

Design teams should enumerate as many potential privacy risks as they can, based on their privacy risk analysis; then assign values for L, D, and C to each risk; prioritize the risks and choose which ones to address; and finally determine solutions that address them. There are, however, several caveats here. First, this approach only expresses what factors should be taken into account. The values of these factors will have to be judgment calls based on the best knowledge that the design team has, much like the severity rating for a heuristic evaluation [29].

Second, this approach does not address extreme cases (of which there will always be many with respect to privacy). Rather, it looks at risks that are foreseeable and significant, with the expectation that design teams should design applications that protect against these more obvious kinds of risks.

Third, the estimation for damage D should take into account the scale of potential damage. For example, there would be a different level of damage if a system that stored personal information about a dozen individuals was compromised, compared to one that stored personal information about thousands of individuals.

Fourth, the estimation for cost C should take into account the burden to the design team and to the end-user. It would be infeasible for a design team, for example, to deploy a version of web browser cookies that is more privacy-sensitive, since so many other web sites and so many web browsers already use the current version of cookies. This would be something better done by the Internet community as a whole rather than by a single group of designers. Similarly, it would be unrealistic to require a data sharer to pre-configure all options correctly on a new device to use it in a privacy-sensitive manner. This would be something better done by the design team.

It is important to note that the utility of this cost-benefit analysis comes not so much from accurate and precise values, but from having the design team think through the issues of likelihood, damage, and cost, and coming up with solutions for mitigating or for helping end-users managing risk.

After prioritizing the privacy risks (based on likelihood and damage), it is useful to think about how to manage those risks. We present a series of questions to help work out potential solutions.

Managing Privacy Risks
- How does the unwanted disclosure take place? Is it an accident (for example, hitting the wrong button)? A misunderstanding (for example, the data sharer thinks they are doing one thing, but the system does another)? A malicious disclosure?
- How much choice, control, and awareness do data sharers have over their personal information? What kinds of control and feedback mechanisms do data sharers have to give them choice,

control, and awareness? Are these mechanisms simple and understandable? What is the privacy policy, and how is it communicated to data sharers?

- What are the default settings? Are these defaults useful in preserving one's privacy?
- In what cases is it easier, more important, or more cost-effective to *prevent* unwanted disclosures and abuses? *Detect* disclosures and abuses?
- Are there ways for data sharers to maintain plausible deniability?
- What mechanisms for recourse or recovery are there if there is an unwanted disclosure or an abuse of personal information?

We discuss each of these questions below.

How does the unwanted disclosure take place?
One issue to consider is how an unwanted disclosure happens, as it suggests different ways of addressing the issue. For example, if it is an accident, such as hitting the wrong button, then this suggests that a revised user interface may be needed. If it is a misunderstanding along the lines of a mismatched mental model, then this suggests that the user interface needs to provide better feedback. If it is a maliciously exploited disclosure, then there probably needs to be better detection mechanisms and ways of preventing such disclosure again in the future.

How much choice, control, and awareness do data sharers have?
This question looks at how data sharers interact with and understand the system. There are many common interaction design issues here, including providing useful and usable controls (for example, the invisible mode found in messenger applications or the option to turn a system off) and providing useful feedback that is not overwhelming (for example, simple notifications).

It is also useful here to think about what the privacy policies are (if a service is being designed) and how these policies will be communicated to data sharers. Using existing interaction design patterns may be useful if such patterns exist, for example having a privacy policy link at the bottom of a web page [39].

What are the default settings?
Previous research suggests that most users do not change application settings from the defaults [27, 31]. It is important to ensure that the defaults will be "right" for the majority of cases, so that the level of personal information shared is not excessive. The original PARCTab system is a negative example of this, as a data sharer's location was visible to anyone by default.

Is it better to prevent unwanted disclosures or detect them?
In some cases, it is better to prevent disclosures, especially if the potential for damage is high. In other cases, however, it may be easier to detect unwanted disclosures instead.

Povey describes an example of the latter with respect to medical emergencies at hospitals, which he calls optimistic access control [33]. The observation here is that there will always be unforeseen situations which cannot be predicted beforehand. In cases where flexibility is important, it may be more useful to allow greater access and have better auditing for detecting misuses. In other words, it is better to have users ask for forgiveness rather than permission. Grudin and Horvitz have argued that in many cases, this approach is easier for people to understand and manage [17].

AT&T's mMode demonstrates another example of optimistic access control with their Find Friends application [6]. A data sharer first sets up a buddy list that lists which friends can request

his current location. Afterwards, that data sharer's friends can always make such a request, but are informed that the data sharer will get a notification on each request. This approach provides a form of social visibility that, in many cases, is sufficient to prevent abuses.

The key factors here are the probability of an unwanted disclosure and the subsequent damage. In cases where either of these values are relatively low (for example, environments with a high level of trust), then detecting errors may be a better approach.

Are there ways for data sharers to maintain plausible deniability?
Plausible deniability is a very powerful mechanism in maintaining social relationships with others. A good example of this is with mobile phones. If a person does not answer a mobile phone call, it could be for technical reasons—such as being outside of the service range, not having the phone with them, or that the phone is off—or for social reasons, such as being busy or not wanting to talk to the caller right now. The result is that the person being called has a simple model for protecting their privacy, while the caller cannot tell why that person is not answering [43].

What mechanisms for recourse or recovery are there?
This question looks at what happens if there has been an unwanted disclosure of personal information. The first issue is to consider what options a data sharer has for recourse, how a data sharer can be made aware of these options, and how the design team can facilitate those options. Again, we use Lessig's framework of technical, market, social, and legal forces. Examples of technical recourse include blocking a data observer, providing an invisible mode, or going to a web page and changing an incorrect option. An example of market recourse is complaining to a company or switching to another service. An example of social recourse is asking someone to stop. An example of legal recourse is to sue an offending party.

An important issue here is to consider how a data sharer can be made aware that an unwanted disclosure has occurred. It could be a notification or a log describing what information went out, or a negative side effect, such as spam.

Before coming up with a solution, it is also useful to consider whether or not a data sharer actually cares that personal information was disclosed. In some cases, the disclosure could be simply an annoyance rather than a serious risk, such as a roommate discovering that a data sharer is out shopping. It is useful to consider this possibility, to avoid over-designing and over-engineering a system.

4. USING THE PRIVACY RISK MODEL
In this section, we describe two case studies in using the privacy risk model outlined above in our work in developing privacy-sensitive ubiquitous computing systems. Both case studies also used lo-fi prototypes and interviews to help inform the design.

Case Study 1 – Location-enhanced Instant Messenger
Our first application combines AT&T Wireless's Find Friends feature with instant messenger. A data sharer has a single buddy list, and can choose to share her location with everyone on her buddy list through a status message (ex. "at home" or "near Soda Hall") or allow others to freely query her current location in a manner similar to Find Friends. Data sharers can also use an invisible mode to prevent any data observers from seeing them. Below is an abbreviated privacy risk analysis of our design.

Who are the users?
The users are people who are willing to share snapshots of their location information with their significant other, family, friends, or possibly co-workers. The service provider is also a third-party data observer.

What kinds of personal information are shared?
Data observers will be able to see the closest place (at roughly the street level) to the data sharer's location.

What are the relationships between data sharers and data observers?
While a user could choose to share their location with anyone, the most common relationships will be social ones with a relatively high level of trust. This application might also be used in work environments, where the trust levels might not be quite so high. The main concerns here will likely be over-monitoring.

How much information is shared?
Data sharers share their location in a discrete one-time manner. Data observers can repeatedly query for a location, but are informed that data sharers will see each of these requests. This kind of notification and awareness by both parties creates a social backpressure that can prevent many kinds of abuses.

How is the personal information collected or shared?
Location information is collected on one's personal device through the use of GPS or 802.11b access point beacons [35]. Thus, location is initially captured in a privacy-sensitive manner.

Are there malicious data observers?
Some potential malicious observers include abusive spouses, stalkers, and spammers. To a large extent, these malicious observers can be managed since location information starts with the data observer, since the data observer has to explicitly have a data observer on a buddy list, and since the data observer gets a notification each time their location information is requested.

Given this analysis, we believe some privacy risks will be:
- Over-monitoring by family or friends (for example, a mother that is over-concerned about her young teenager)
- Over-monitoring by co-workers and supervisor
- Being found by a malicious person (such as a stalker)

In the interests of space, we only examine the first risk, over-monitoring by family or friends. The likelihood of this happening will vary significantly depending on the individual data sharer. It is probably better to err on the conservative side, so we assign this a likelihood value of high. The damage that could occur is more likely to be embarrassment or annoyance rather than immediate danger to one's safety, so we assign a damage value of medium.

Next, we look at how these risks might be managed.

How much choice, control, and awareness do data sharers have?
Data sharers set their own status—invisible, offline, or online. Offline means exiting the entire application. Online is when the data sharer has complete access to the functionality of the application. Invisible mode means data sharers can see their online buddies, but online buddies cannot see the data sharers. Also, any significant events such as requesting a buddy's location or received location requests are stored in history logs.

What mechanisms for recourse or recovery are there?
A data sharer can ask someone directly why she is repeatedly querying for a data sharer's location. A data sharer can also remove a buddy from their buddy list, preventing that person from retrieving future location information.

The buddy list, invisible mode, and notifications are fairly simple to implement, so we assign cost a value of low. Thus we have a high likelihood, a medium damage, and a low cost. Since the cost is low relative to the potential risk, it suggests that we should have these features. Again, we note that it will be difficult to estimate accurate values, especially given a wide range of users and contexts of use. However, we argue that it is far better to have the design team consider these issues using their best judgment and to discuss what is reasonable, than not doing so.

We also complemented our analysis above through interviews and low-fi prototypes with twenty people, to help inform our risk analysis and to provide a greater understanding of potential privacy concerns with such a system. Some interviewees were concerned that their location could be misinterpreted. For example, a student can go to a café either to meet someone or to study. The most common concern was over-monitoring by friends and family. Indeed, several participants were worried that their location would be constantly checked. One participant had a concern with his girlfriend using the application, "She would get suspicious. She would use it all the time." Another primary concern was the social pressures created from using the system, such as choosing who to authorize and who to ignore. Because all online buddies would have equal access to location, one participant noted, "It's hard to say no [to authorizing people…] because you would be mean. You get a long list of people you don't want to be friends [with…] and you might regret it later on and you have to be put yourself on invisible. Then [that is] just another hassle about it."

We also created lo-fi prototypes of the location-enhanced messenger and tested them with three users. This evaluation revealed that potential users view the location feature as separate from the normal mode of instant messaging. While it is "safe" to communicate with random strangers through instant messenger, the same is not true with respect to location.

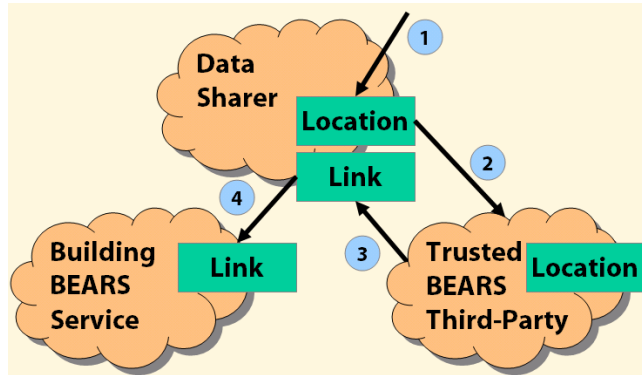## Case Study 2 – BEARS Emergency Response Service
Enhanced 911 lets users share their location with dispatchers when making emergency calls on mobile phones. One's location is only transmitted to dispatchers when the call is actually made. While there are many advantages to E911, one downside is that it is a discrete push system. There are no easy ways of getting a person's current or last-known location in known emergencies, for example, an earthquake, a building fire, or a kidnapping.

BEARS is a system we are developing to handle these cases. There are two tensions to balance here. On the one hand, we want location information to be highly available in the case of emergencies. On the other hand, emergencies are rather rare, and so we also want some guarantees that location information will be used exclusively for emergencies and for no other purposes.

BEARS works by having a trusted third-party store one's location information for use in case of emergency. This third party can be a friend or even a paid service whose business model is predicated on providing location information only in the event of emergencies. Such services already exist with respect to one's medical information, the best known of which is MedicAlert [1]. These services would have a significant market incentive to use location information only for stated purposes and possibly a legal obligation as well.

Figure 1 shows an example of how BEARS can be used in buildings to keep track of who is in the building and where they are for emergency response purposes. First, a data sharer obtains his location. He sends his location to the trusted third party, which gives him one or more named links back to this data (multiple links can be used to eliminate unique identifiers, if a data sharer wishes). The data sharer can then share this link with others, such as a building. In case of emergencies, the link can be traversed, with last-known location information being retrieved.

Figure 1. An example setup of the BEARS emergency response service. A data sharer obtains their location (1) and shares it with a trusted third-party (2). The end-user gets a link (3) that can be sent to others, in this case to a building (4). If there is an emergency, responders can traverse all known links, getting up-to-date information about who is in the building (with the trusted third-party notifying data sharers what has happened).

Below is an abbreviated privacy risk analysis of BEARS:

Who are the users of the system?
The data sharers are people who need to share their location with emergency response personnel. The data observers are the trusted third party, any deployments of BEARS (e.g., a building), and emergency responders.

What are the relationships between data sharers and data observers?
In the paid service scenario, the relationship is a market and a legal one.

Are there malicious data observers?
Some potential malicious observers include spammers and stalkers. If they knew a data sharer was using BEARS, the malicious observer could contact the trusted third party, act as emergency responders and obtain authorization for location information.

How much information is shared?
The trusted third party gets continuous updates of a data sharer's location. This information is not disclosed to the emergency responder until the trusted third party has authorized it.

How is personal information stored?
The location information is stored on the third party's server but is inaccessible until they have authorized its release. A building where the emergency occurred will only have a link that points to the data sharer's location.

How much choice, control, and awareness do data sharers have?
When setting up the service, data sharers choose who they want as their proxy or the trusted third party. When a data sharer enters a building with emergency response service, they can choose to

either "always activate", "activate one time" or "never activate" that BEARS service.

Given this analysis, we identify two likely privacy risks:
• A malicious observer pretends that there is an emergency to get someone's location information
• A false emergency, where the location of everyone in that building was divulged

In the interests of space, we only examine the first privacy risk. This is a serious concern, because it undermines the use of location information for emergency response purposes only. The likelihood of this happening is quite high, and the damage would also be high, to the data sharer and to the trusted third party.

Our interviews with the same twenty people as in the previous case study helped inform our analysis. Interviewees pointed out some of the concerns noted above, such as that it had the potential be improperly used when not in an emergency situation. Although some interviewees indicated a strong desire for such a system, others were worried that the government or other authorities would use it for other purposes. One participant asked, "Do [the authorities] watch you every minute?" This issue is partially addressed by the fact that this is an opt-in system, where data sharers voluntarily choose to use it. The local server "controls" BEARS, where it determines whether a situation is an emergency. During the evaluation of a lo-fi prototype, one participant echoed the sentiment, "[I] want to have control whether information is being sent or not." Predominantly, prevention of disclosure to malicious data observers is the main solution.

One way of managing these risks is to provide better mechanisms to ensure that only authorized emergency responders see location information. For example, MedicAlert is setup to only accept phone calls from hospitals (which can be identified in advance). One could imagine a similar situation with BEARS, where only a phone call from a police station or fire station could allow disclosure of location information. Furthermore, if no strong social relationship existed between the data sharer and the third party, the third party could sign a legal agreement not to disclose any personal information except for emergency purposes.

5. CONCLUSIONS
In this paper, we argued for using privacy risk models as a way of refining privacy from an abstract concept into a set of concrete concerns for a specific domain and community of users. We introduced a specific privacy risk model for personal privacy in ubiquitous computing, and described two case studies of how we used privacy risk models in designing two ubiquitous computing applications. Combined with interviews and lo-fi prototypes, the privacy risk model has helped us identify a number of privacy risks in two ubicomp applications in development and come up with solutions for managing those risks.

As we noted earlier, just as no security threat model is perfect, and just as no task analysis is perfect, no privacy risk model is perfect. No analysis can predict every potential use of personal information. What a privacy risk model does do is help designers consider the specific community of users, potential risks and benefits, control and feedback mechanisms, and means for recourse and recovery, In other words, it helps them design for a *reasonable* level of privacy.

Privacy is a difficult design issue deeply enmeshed in social, legal, technical, and market forces. Privacy risk models are a step forward in helping us understand and design for these issues as we continue pushing forward into augmented environments.

6. ACKNOWLEDGMENTS

7. REFERENCES
1. MedicAlert. http://www.medicalert.org
2. United States v. Carroll Towing Co. 1947.
3. Adams, A. Multimedia Information Changes the Whole Privacy Ball Game. In Proceedings of *Computers, Freedom, and Privacy*. Toronto, Canada: ACM Press. pp. 25-32 2000.
4. Agre, P.E. and M. Rotenberg, *Technology and Privacy: The New Landscape*. Cambridge MA: MIT Press, 1997.
5. allnurses.com - Nursing Discussion Board for Nurses Archive, Nurse Tracking Devices: Whats Your Opinion? http://allnurses.com/forums/showthread/t-8012.html
6. AT&T, AT&T Wireless mMode - Find Friends. http://www.attwireless.com/mmode/features/findit/FindFriends/
7. Barkhuus, L. and A.K. Dey. Location-based services for mobile telephony: a study of users' privacy concerns. In Proceedings of *INTERACT 2003, 9th IFIP TC13 International Conference on Human-Computer Interaction*. pp. To appear 2003.
8. BBC News, Radio tags spark privacy worries. http://news.bbc.co.uk/1/hi/technology/3224920.stm
9. Bellotti, V. and A. Sellen. Design for Privacy in Ubiquitous Computing Environments. In Proceedings of *The Third European Conference on Computer Supported Cooperative Work (ECSCW'93)*. Milan, Italy: Kluwer Academic Publishers 1993.
10. Brin, D., *The Transparent Society*. Reading, MA: Perseus Books, 1998.
11. Doheny-Farina, S., The Last Link: Default = Offline, Or Why Ubicomp Scares Me, *Computer-mediated Communication*, vol. 1(6): pp. 18-20, 1994.
12. Etzioni, A., *The Limits of Privacy*. New York: Basic Books, 1999.
13. Feinman, J.M., *Law 101*. Oxford, England: Oxford University Press, 2000.
14. Felten, E., DRM, and the First Rule of Security Analysis. 2003. http://www.freedom-to-tinker.com/archives/000317.html
15. Garfinkel, S., *Database Nation: The Death of Privacy in the 21st Century*: O'Reilly & Associates, 2001.
16. Grudin, J., Groupware and Social Dynamics: Eight Challenges for Developers, *Communications of the ACM*, vol. 37(1): pp. 92-105., 1994.
17. Grudin, J. and E. Horvitz, Presenting choices in context: approaches to information sharing. 2003: Workshop on Ubicomp communities: Privacy as Boundary Negotiation. http://guir.berkeley.edu/pubs/ubicomp2003/privacyworkshop/papers.htm
18. Hackos, J.T. and J.C. Redish, *User and Task Analysis for Interface Design*. Hoboken, NJ: John Wiley & Sons, 1998.
19. Harper, R.H.R., *Why Do People Wear Active Badges?* Technical Report EPC-1993-120, Rank Xerox, Cambridge 1993.
20. Harter, A. and A. Hopper, A Distributed Location System for the Active Office. *IEEE Network*, 1994. **8**(1).
21. 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*. Göteborg, Sweden. pp. 176-193 2002.
22. Langheinrich, M. Privacy by Design - Principles of Privacy-Aware Ubiquitous Systems. In Proceedings of *Ubicomp 2001*. Atlanta, GA. pp. 273-291 2001.
23. Lederer, S., J.I. Hong, A. Dey, and J.A. Landay, Personal Privacy through Understanding and Action: Five Pitfalls for Designers. *Personal and Ubiquitous Computing*, 2004. To Appear.
24. Lederer, S., J. Mankoff, and A. Dey, Towards a Deconstruction of the Privacy Space., Workshop on Privacy In Ubicomp'2003: Ubicomp communities: privacy as boundary negotiation. http://guir.berkeley.edu/pubs/ubicomp2003/privacyworkshop/papers/lederer-privacyspace.pdf
25. Lederer, S., J. Mankoff, and A.K. Dey. Who Wants to Know What When? Privacy Preference Determinants in Ubiquitous Computing. In *Extended Abstracts of CHI 2003, ACM Conference on Human Factors in Computing Systems*. Ft Lauderdale, FL. pp. 724-725 2003.
26. Lessig, L., *Code and Other Laws of Cyberspace*. New York NY: Basic Books, 1999.
27. Mackay, W.E. Triggers and barriers to customizing software. In Proceedings of *ACM CHI '91 Human Factors in Computing Systems*. New Orleans, LA 1991.
28. Marx, G., Identity and Anonymity: Some Conceptual Distinctions and Issues for Research, in *Documenting Individual Identity: The Development Of State Practices In The Modern World*, J. Caplan and J. Torpey, Editors. Princeton University Press, 2001.
29. Nielsen, J., *Usability Engineering*. Boston, MA: Academic Press, 1993.
30. Nodder, C., Say versus Do; building a trust framework through users' actions, not their words. Workshop on Privacy In Ubicomp'2003: Ubicomp communities: privacy as boundary negotiation. http://guir.berkeley.edu/pubs/ubicomp2003/privacyworkshop/
31. Palen, L., Social, Individual and Technological Issues for Groupware Calendar Systems. *CHI Letters: Human Factors in Computing Systems, CHI 99*, 1999. **2**(1): p. 17-24.
32. Palen, L. and P. Dourish, Unpacking "Privacy" for a Networked World. *CHI Letters*, 2003. **5**(1): p. 129-136.
33. Povey, D. Optimistic Security: A New Access Control Paradigm. In Proceedings of *1999 New Security Paradigms Workshop* 1999.
34. Saltzer, J.H. and M.D. Schroeder, The Protection of Information in Computer Systems. *Proceedings of the IEEE*, 1975. **63**(9): p. 1278-1308.
35. Schilit, B.N., *et al.* Challenge: Ubiquitous Location-Aware Computing. In Proceedings of *The First ACM International Workshop on Wireless Mobile Applications and Services on WLAN Hotspots (WMASH '03)*. San Diego, CA: ACM Press. pp. To Appear 2003.
36. Schilit, B.N., J.I. Hong, and M. Gruteser, Wireless Location Privacy Protection, *Computer*, vol. 36(12): pp. 135-137, 2003.
37. Sloane, L., Orwellian Dream Come True: A Badge That Pinpoints You, *New York Times* pp. 14, 1992.
38. Talbott, S., The Trouble with Ubiquitous Technology Pushers, or: Why We'd Be Better Off without the MIT Media Lab. 2000. http://www.oreilly.com/people/staff/stevet/netfuture/2000/Jan0600_100.html
39. van Duyne, D.K., J.A. Landay, and J.I. Hong, *The Design of Sites: Principles, Processes, and Patterns for Crafting a Customer-Centered Web Experience*. Reading, MA: Addison-Wesley, 2002.
40. Weiser, M., R. Gold, and J.S. Brown, The Origins of Ubiquitous Computing Research at PARC in the Late 1980s. *IBM Systems Journal*, 1999. **38**(4): p. 693-696.
41. Westin, A.F., *Privacy and Freedom*. New York NY: Atheneum, 1967.
42. Whalen, J., You're Not Paranoid: They Really Are Watching You, *Wired Magazine*, vol. 3(3): pp. 95-85, 1995.
43. Woodruff, A. and P.M. Aoiki. How Push-to-Talk Makes Talk Less Pushy. In Proceedings of *ACM SIGGROUP Conf. on Supporting Group Work (GROUP '03)*. Sanibel Island, FL. pp. 170-179 2003.