An Architecture for Privacy-Sensitive Ubiquitous Computing

By

Jason I-An Hong

B.S. (Georgia Institute of Technology) 1997

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Computer Science

in the

GRADUATE DIVISION

of the

UNIVERSITY OF CALIFORNIA, BERKELEY

Committee in charge:
Professor James A. Landay, Chair
Professor John C. Canny
Professor Paul K. Wright
Professor Terry Winograd

Spring 2005

The dissertation of Jason I-An Hong is approved:

Chair Date

Date

Date

Date

University of California, Berkeley

Spring 2005

An Architecture for Privacy-Sensitive Ubiquitous Computing

by

Jason I-An Hong

Abstract

An Architecture for Privacy-Sensitive Ubiquitous Computing

by

Jason I-An Hong

Doctor of Philosophy in Engineering

Electrical Engineering and Computer Sciences

University of California, Berkeley

Professor James A. Landay, Chair

Privacy is easily the most often-cited criticism of ubiquitous computing (ubicomp), and may be the greatest barrier to its long-term success. However, developers currently have little support in designing system architectures and in creating interactions that are effective in helping end-users manage their privacy.

This dissertation provides three key contributions towards ameliorating this problem. The first contribution is an extensive analysis of end-user privacy needs, which were gathered through a variety of techniques, including interviews, surveys, synthesis of previously reported experiences with ubiquitous computing, as well as examination of proposed and existing data privacy laws.

The second contribution is an analysis of interaction design for ubicomp privacy. Informed by examining over 40 different user interfaces for privacy, we describe common user interface pitfalls as well as ways of avoiding those pitfalls.

The third contribution is a system architecture that embodies the two analyses above. We present Confab, a toolkit that facilitates the construction of privacy-sensitive ubicomp applications by providing a customizable framework for capturing, processing, and sharing personal information in a privacy-sensitive manner. From a system architecture perspective, Confab emphasizes two key ideas. The first is separating ubicomp applications into the physical / sensor layer, the infrastructure layer, and the presentation layer, with each of these being responsible for managing and providing privacy protection for different aspects of the flow of personal information. The second key idea is to structure the system so that end-users have personal information captured, stored, and processed on their computers as much as possible, and are provided better user interfaces for managing the flow of personal information to others.

Confab currently comes with extensions specifically for managing location privacy in applications built within this framework. We also present an evaluation of this toolkit based on building three applications and performing user studies of those applications.

Professor James A. Landay
Dissertation Committee Chair

To all my family, to all my friends:

the journey has been long,

but made all the more worthwhile because of all of you.

To the bureaucracy at Berkeley's Grad Division Office:

for giving me penetrating insights about Kafka,

may you live in interesting times.

Table of Contents

List of Figures

List of Tables

Acknowledgments

I would like to start by thanking Mark Weiser. I only had the fortune of meeting him once before his untimely passing, but the passion behind his words has inspired me and many others to pursue this seemingly impossible dream. We'll do our best to guide ubicomp into becoming something we can all be proud of.

Chris Beckmann, Jeff Heer, and Alan Newberger designed and implemented the liquid distributed querying system on top of Confab, compelling me to fix many bugs and clarify the API, ultimately making Confab all the more useful.

Jennifer Ng worked on the end-user interviews and on transcribing audio notes. She also worked with Eric Chung and Madhu Prabaker in collecting place names and WiFi data around Berkeley. All three of you are among the best undergrads I have had the pleasure of working with, I look forward to watching you blossom and grow.

Xiaodong Jiang and Scott Lederer provided a lively intellectual forum for discussing privacy issues. Scott provided the survey data which comprises part of the analysis in Chapter 2 and Appendix C, and took the lead in developing the pitfalls in user interfaces for privacy, which is presented in Chapter 4 of this work.

This work has been vastly improved due to feedback from many different people, including Gregory Abowd, John Canny, Anind Dey, Keith Edwards, Marti Hearst, Jen Mankoff, Deirdre Mulligan, Bill Schilit, Doug Tygar, Terry Winograd, and Paul Wright. Special thanks to Bill Schilit for guiding me in my research since I started grad school, and for always challenging me to consider the bigger picture. Special thanks also to John Canny and Doug Tygar, for helping me push this work forward and for always providing insightful advice right when I really needed it.

I would like to express my appreciation to everyone in GUIR for providing a rich and animated intellectual community. In particular, I would like to thank Kat(i)e Everitt, Wai-ling Ho-ching, Scott Klemmer, Yang Li, Chris Long, Anoop Sinha, Sarah Waterson, and honorary members David Oppenheimer and Andy Begel. I would also like to express my gratitude to my many office mates over the years. To James Lin, we've been in the same office for seven years, thankfully one year shy of common-law marriage in California. To Richard Davis, Mark Newman, Jack Chen, Francis Li, Hesham Kamel, Xiaodong Jiang, and honorary office member Rich Vuduc, I have countless memories of late night Abba music, geocaching hunts, crane origami, Dance Dance Revolution, ramen noodle runs, inappropriate Simpsons' quotes, near-car-crash-causing impressions of certain professors, innumerable Birkball breaks…now that I think about it, how did we ever get any work done?

Special thanks to my friends for dragging me away from work to have fun, for putting up with my lame excuses, and for generally keeping me sane (though I am sure many of you would argue that). I promise that if I ever get around to writing that great American novel, I will be sure to include all of you. My deepest appreciations go to my high school buddies John (blow up a cow?) Bulalacao, Chris Crosby, Adam King, and Tony Soong, as well as my friends Trinh Carpenter (I live vicariously thru you), Allison Chu, Jim Dooley, Idris Hsi, Larry Hsieh, Elaine Huang, Jessica Kao, Eugene Liang, Wes Parrish, Gaius “I speak to my cat in Latin” Stern, Quan Tran, and Khai Truong. To Leila Takayama, I have greatly appreciated your warmth and humor over the past few years. Thanks for being a patient friend to my quirky sense of humor and insane schedule. To Chris Yueh, I think you know too many embarrassing stories about me, so thanks for keeping quiet over these many years. More seriously, though, I think we’ve almost been through it all together, and I couldn’t have asked for a better friend.

To my advisor James Landay, I lost count of the number of times you went to bat for me. Thanks for setting the bar high, and for always having faith in me, even when I did not. I couldn’t have done it without you.

Lastly, I would like to thank my family. Dad, mom, Jerry, Cordelia, thanks for keeping me grounded and making sure I wasn’t too absent-minded in taking care of myself. And to my little nephew Ethan and the soon-to-be-born twins, it will be many years before any of you are old enough to read this, but I want to thank all of you for helping to put my life in perspective and for reminding me what this is all for. This chapter of my life is coming to a wistful close, but I have many more snowfalls to see, many more friends to meet, and many more dreams to chase. I’m lucky I have all of you to share it with.

This work has been supported in part by NSF (IIS-0205644), DARPA (N66001-99-2-8913), an Intel fellowship, a Siebel Scholar fellowship, and PARC.

Part I

Motivation for Privacy-Sensitive Ubiquitous Computing

1 Introduction

Over the past decade, there has been an increasing trend towards integrating sensing, communication, and computation into the physical world. No longer restricted to the office desktop, computers are becoming embedded in all aspects of our everyday lives, varying from electronic toys to smart cars, from augmented classrooms to intelligent homes. These computers are also becoming increasingly aware of the environments and situations in which they are used, using factors as simple as the current humidity and light level to as complex as who is using the computer and where it is being used. This push towards *ubiquitous computing* [147] offers tremendous gains in coordination, safety, and efficiency in domains as diverse as real-time monitoring of soil conditions [28], helping patients with Alzheimer's disease [127], and support for emergency responders [83].

The fundamental problem, however, is that these same technologies also introduce many new privacy risks, often at a rate faster than legal mechanisms and social norms can adapt. Ubiquitous computing technologies change the privacy landscape by dramatically lowering the cost of collection, making it easy to gather and share a wide range of data about individuals, all in real-time and in a manner that is machine readable and searchable. The risks posed by ubicomp technologies range from everyday ones—such as intrusions from overprotective parents and overzealous marketers—to extreme ones, such as threats to civil liberties by governments as well as dangers to one’s personal safety by stalkers, muggers, and domestic abusers. Numerous interviews (e.g. [17, 71, 86]), essays (e.g. [46, 139, 148]), books (e.g. [10, 26, 59]), and negative media coverage (e.g. [134, 150]) have repeatedly described peoples’ concerns regarding the strong potential for abuse, general unease over a potential lack of control, and overall desire for privacy-sensitive systems. In some cases, these concerns have even led to outright rejection of systems [71, 123], strongly suggesting that privacy may be the greatest barrier to the long-term success of ubiquitous computing technologies.

The difficulty here is that little work has been done to address the issue of ubicomp privacy. The large majority of previous work has been on traditional computing systems and has tended to focus on providing anonymity or on keeping personal information and messages secret from hackers, governments, and faceless corporations. While anonymity and secrecy are clearly important, they only address a relatively narrow aspect of privacy and do not cover the many situations in everyday life where people *do* want to share information with others. 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 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.

The point is that, rather than being a single monolithic concept, privacy is a fluid and malleable notion with a range of trust levels and needs. *Our goal here is to empower people with choice and informed consent, so that they can choose to share the right information, with the right people and services, in the right situations.* As Weiser noted, “The problem, while often couched in terms of privacy, is really one of control. If the computational system is invisible as well as extensive, it becomes hard to know what is controlling what, what is connected to what, where information is flowing, how it is being used… and what are the consequences of any given action” [148].

The key problem this dissertation addresses is that it is difficult to create privacy-sensitive ubicomp applications. To address this, we present the Confab toolkit, which, based on an extensive analysis of end-user needs and interaction design for ubicomp privacy, facilitates the construction of high-quality privacy-sensitive ubicomp applications.

Our focus here is primarily on *personal privacy*, which is the processes by which individuals selectively disclose personal information—such as email address, shopping history, or location—to organizations and to other people.[1] However, even if designers and developers are interested in building and deploying privacy-sensitive ubicomp systems, they currently have little guidance or support for doing so.

[1] This is in contrast to what we call *organizational privacy*, where an organization (such as a company or a government) is concerned about how information about customers or citizens is managed.

Towards this end, we address three related problems. The first problem is that it is hard to *analyze* privacy needs. Currently, there is a great deal of speculation about what privacy concerns people have with respect to ubiquitous computing, but little actual data that can be used to inform the design of such systems. To address this, we present an analysis of end-user privacy needs gathered through a variety of techniques, including interviews, surveys, investigation of previously reported experiences with ubiquitous computing, as well as an examination of proposed and existing data privacy laws.

The second problem is that it is hard to *design* effective user interfaces for privacy. To address this, we present an analysis of interaction design for ubicomp privacy. Informed by examining over 40 different user interfaces for privacy, we describe common user interface pitfalls as well as ways of avoiding those pitfalls.

The third problem is that it is hard to *implement* privacy-sensitive systems. To address this, we present the Confab toolkit, which draws its requirements from the two sets of analyses described above, and provides a framework for building privacy-sensitive ubicomp applications. This framework can be partitioned into three independent layers for managing privacy, including:

- the *physical / sensor layer*, which is responsible for initially capturing personal information;
- the *infrastructure layer*, which is responsible for storing and processing personal information; and

- the *presentation layer*, which is responsible for providing user interfaces to give end-users greater control and feedback over their personal information.

Confab currently comes with extensions built within this framework specifically for managing location privacy. We also present an evaluation of this toolkit based on building three applications and performing user studies of those applications.

To provide a deeper understanding of the importance of privacy as well as the difficulties involved in building privacy-sensitive ubicomp applications, we present a brief history of ubiquitous computing and its tensions with privacy.

1.1 A Historical Perspective on Privacy and Ubiquitous Computing

Ubiquitous computing originated in the late 1980s as a reaction to what was seen as wrong with the personal computer [148]. Researchers at Xerox PARC saw computers as too complex to use, too demanding of our attention, and too isolating from other people and activities. At the same time, advances in sensors, wireless networking, and devices of all form factors were enabling sensing, computation, and communication to be integrated into the physical world at large. At a philosophical level, ubiquitous computing was continuing the trend of extending our senses, vastly expanding our ability to see, remember, and communicate with one another. The key insight was that this should be done at such a vast scale and in such a deep and fundamental manner that we would no longer be consciously aware of the technology, freeing us to focus on goals rather than the means. As science fiction author Vernor Vinge has stated, we will soon reach a point where the combination of powerful processors, limitless data-storage capacity, ubiquitous sensor networks, and deeply embedded user interfaces will create a bond between human and machine "so intimate that users may reasonably be considered superhumanly intelligent" [62].

A key project in the initial foray into ubiquitous computing was the PARCTab system [145]. PARCTabs were small pen-based devices that connected to a local area network through wireless gateways set up in each room. PARCTabs were also designed to periodically beacon out a signal, allowing a network service to determine what room each PARCTab was currently in. A primary insight pioneered by the developers of the PARCTab system was that contextual information—in this case, the user's identity, location, and nearby people and resources—could be leveraged to provide useful services tailored to the current situation. For example, a user could create contextual reminders (e.g., "next time I'm in the library" or "when I'm back at my desk") [130], have information that user was interested in be automatically shown on nearby public displays (e.g., hockey scores) [130], or have an electronic diary that automatically created entries of places gone and people seen [91]. Researchers at Xerox PARC also experimented with Active Badges [73], wearable badges that provided indoor location tracking and identification.

However, privacy concerns were raised almost immediately and dogged these projects throughout their existence. Many people at Xerox PARC had visceral and highly emotional responses to the research. One researcher said:

> “Do I wear badges? No way. I am completely against wearing badges. I don’t want management to know where I am. No. I think the people who made them should be taken out and shot... it is stupid to think that they should research badges because it is technologically interesting. They (badges) will be used to track me around. They will be used to track me around in my private life. They make me furious.” [71]

The news media also jumped immediately on the privacy risks posed by these technologies, publishing headlines such as “Big Brother, Pinned to Your Chest” [39] and “Orwellian Dream Come True: A Badge That Pinpoints You” [134]. Ubiquitous computing was seen less as something that could help people in their everyday lives and more as a pervasive surveillance system that would further cement those already in positions of power. Others outside of the news media made similar observations. For example, communications professor Stephen Doheny-Farina published an essay entitled “Default = Offline, or Why Ubicomp Scares Me” [46]. Howard Rheingold summarized it best when he observed that ubiquitous computing technologies “might lead directly to a future of safe, efficient, soulless, and merciless universal surveillance” [124].

Less sensationalistic, though no less instructive, is the fact that some ubiquitous computing technologies have already been rejected even in cases where they might provide value for end-users. For example, some hospitals have their nurses wear locator badges (essentially Active Badges) that can be used to facilitate coordination and protect nurses from spurious patient claims, for example, not getting any service. However, in many cases, these locator badges have led to increased friction between workers and employers, as they were perceived by nurses as a surreptitious surveillance system [12-14]. In at least two separate cases, nurses outright refused to wear the locator badges [31, 123]. *The main point here is that ubicomp technologies are often perceived as violating expected norms and boundaries surrounding privacy, posing a significant barrier to entry even in systems that provide value to end-users.*

Interestingly, privacy was explicitly mentioned as a key research issue in “The Computer for the 21st Century” [147], the seminal paper that introduced ubiquitous computing in 1991. Privacy has also been consistently raised as a crucial issue to the long-term success of ubiquitous computing in numerous other research papers, workshops, and panels (for example, [37, 38, 128, 135]). There has also been a growing awareness in industrialized nations of privacy issues in general, particularly due to the widespread use of the World Wide Web. For example, an August 2000 poll by The Pew Internet & American Life Project found that 84% of people were very concerned or somewhat concerned about “businesses and people you don’t know getting personal information about you and your family”, and that 79% of people thought that Internet companies should ask people for permission to use personal information [120]. A February 2003 Harris poll found that 69% of those surveyed agreed that “consumers have lost all control over how personal information is collected and used by companies” [140].

Despite this broad consensus, designers and developers currently have little support in developing and deploying privacy-sensitive systems, even if they are committed to doing so. Previous research, such as the PARCTab system [129], the Context Toolkit [45], and iROS [85], provide support for building ubicomp applications, but do not provide features for managing privacy. The result is that privacy is done in an ad hoc manner and often as an afterthought, if at all, leading to applications that end-users may ultimately reject because they are uncomfortable using them or find them intrusive.

Furthermore, the need for privacy-sensitive ubicomp is quickly rising as these technologies become cheaper and easier to deploy. We are starting to see hypothetical privacy risks turn into actual ones. For example, one Connecticut rental car company equipped its vehicles with GPS devices and, after tracking a customer’s van, fined him $450 USD for speeding on three occasions [97]. As another example, in 2002, one hospital has instituted new rules on the number of times nurses can use the restroom in a given month, using cameras and voice recognition technologies to enforce these policies [13].

To summarize, there are three main points here. First, privacy poses a significant barrier to the successful deployment of many ubicomp technologies. Second, even if designers and developers want to construct privacy-sensitive systems, there is currently little guidance or system support for doing so. Third, we are just starting to see privacy abuses from ubicomp technologies, and it is likely that we will see more in the future. It is important to develop and deploy privacy-sensitive ubicomp systems now, so that we can maximize the real benefit of these technologies while minimizing potential and actual risks, before these technologies become so widespread that it becomes difficult or even impossible to change them.

1.2 Challenges in Building Privacy-Sensitive Ubicomp Applications

In this section, we examine why privacy-sensitive ubicomp applications are difficult to build and deploy. From a computer science perspective, the primary difficulty lies in the fact that privacy is not a purely technical issue, but also involves aspects of legislation, corporate policy, and social norms. Furthermore, privacy is a malleable concept in practice, based on individual perceptions of risk and benefit. For example, many people routinely use a credit card to buy goods and services on the Internet because they believe that the convenience of online purchases outweighs the potential cost of such transaction data being misused. This general sense of the difficulties involved led many researchers of ubicomp systems to simply leave privacy as future work, as it was a problem that could never be “solved” in the traditional computer science sense.[2]

However, as Lessig has noted, the way a technology is designed has significant impact on how other forces, including laws, market forces, and social norms, can be brought to bear on the problem [99]. Or, as Bellotti and Sellen put it, a poorly designed system might interfere with social behavior, which could “foster unethical use of the technology… [and be] much more conducive to inadvertent intrusions on privacy” [21].

This is the general philosophy we have taken in this dissertation. More specifically, rather than trying to solve privacy, our focus is on helping people *manage* their personal privacy, empowering them with choice and informed consent so that they can choose to share the right information, with the right people and services, in the right situations. We approach the problem of ubicomp privacy in terms of providing a more solid technical foundation for building applications, as well as better user interface widgets and user interface design guidelines to help end-users manage privacy, giving end-users greater control and feedback over their personal information than previous systems.

[2] In this dissertation, we take a pragmatic technical- and design-oriented view as to why privacy-sensitive ubicomp apps are hard to build and deploy. Other perspectives that are beyond the scope of this dissertation include an economic perspective (such as market incentives for deploying privacy-sensitive apps and price discrimination issues), a public policy perspective (for example, balancing privacy needs with social goals), or a philosophical perspective (for example, that in a liberal democratic society, technologies should be built in alignment with the values of that society)

From an application development perspective, though, there are still several difficulties with this approach. First, it is hard to *analyze* end-user needs for ubicomp privacy. While there is a great deal of speculation and media sensationalization, there is not a great deal of meaningful information that can be used to inform the design of such systems. Second, it is difficult to *design* effective user interfaces for ubicomp privacy. It is not clear what kinds of user interfaces work well and what kinds do not. Third, it is difficult to *build* privacy-sensitive ubicomp applications. It is not clear what abstractions and mechanisms are useful for application developers in managing ubicomp privacy. Furthermore, it takes a high level of technical expertise to design and develop ubicomp systems in general, even without addressing the privacy needs.

1.3 Research Contributions of this Dissertation

This dissertation addresses the three different challenges listed above, each of which forms a primary research contribution of this dissertation. Specifically, these contributions are as follows:

1. To address the first problem, that it is hard to *analyze* end-user needs for ubicomp privacy, we present a *comprehensive set of end-user needs* gathered from a variety of sources. These include scenario-based interviews that we conducted to understand the range of privacy concerns with respect to ubicomp applications, an analysis of freeform comments from a survey on ubicomp

privacy preferences, an investigation of postings on a nurse message board describing experiences using locator systems, a synthesis of previously reported experiences with ubicomp systems, and an examination of proposed and existing privacy laws. This set of needs is useful in informing designers of the range of privacy concerns end-users have with ubicomp systems.

2. To address the second problem, that it is difficult to *design* effective user interfaces for ubicomp privacy, we describe *a set of pitfalls in designing user interfaces for ubicomp privacy, derived from an analysis of over forty different applications for common mistakes still being made*. These pitfalls are useful in informing designers of common user interface mistakes and ways of avoiding those mistakes.
3. To address the third problem, that it is difficult to *build* privacy-sensitive ubicomp applications, we present the *design, implementation, and evaluation of the Confab toolkit*. Based on the set of end-user needs and analysis of user interface pitfalls described above, Confab facilitates the construction of privacy-sensitive ubicomp applications by providing an extensible framework for capturing, processing, and presenting personal information. Confab introduces the idea of protection for ubicomp privacy at the physical, infrastructure, and presentation layers. Confab also introduces an alternative architecture for ubicomp applications, where personal information is captured, stored, and processed as much as possible on computers that end-users have control over,

along with user interfaces for helping end-users make better decisions about disclosures. This is in contrast to previous architectures for ubicomp which have tended to distribute capture, storage, and processing over the network, making it harder for end-users to control the flow of their personal information.

4. To evaluate Confab, we present a number of novel ubicomp applications that were implemented using our toolkit, including a location-enhanced instant messenger, a location-enhanced web proxy, and an emergency response application. We also present results of a user study describing end-users' perceptions of privacy with respect to these applications. In summary, the users all assumed that the location information started with them (regardless of whether this was true or not), understood in general how the applications worked and how to control what location information was disclosed, and were quite interested about using two of the three applications, namely the location-enhanced instant messenger and the location-enhanced web proxy.

This dissertation also introduces several other smaller research contributions, including the following:

- The design, implementation, and evaluation of two reusable graphical user interface components for managing privacy, namely access notifications and the Place Bar

- The design and implementation of privacy tags, which represent another step towards the use of digital rights management for end-user privacy
- The design, implementation, and evaluation of a location-enhanced instant messenger
- The design, implementation, and evaluation of a web proxy that automatically fills in location information on web forms
- The design and implementation of an emergency response application that relies on a trusted third party for managing privacy

It should be noted that Confab is not intended to provide perfect privacy, if there is even such a thing. As noted earlier, privacy ultimately must be managed through a combination of technology, legislation, corporate policy, and social norms [98]. What Confab does provide is a more solid technical foundation for privacy-sensitive ubiquitous computing than previous approaches, making it easier and more feasible for developers to build privacy-sensitive applications for an intended community of users and for companies to offer their services while minimizing the risk to people's privacy

As an analogy, a web design tool can be used to create good as well as bad web sites, but a useful tool will be oriented toward making it easier to create good ones. Similarly, Confab is designed to provide a stronger technical model to ensure that privacy is both a feasible and desirable component of future ubiquitous computing applications.

1.4 Dissertation Outline

Roughly speaking, this dissertation can be divided into three parts. The first part, comprised of this chapter, provides the motivation for privacy-sensitive ubicomp as well as an overview of potential and emerging privacy risks.

The second part describes our analysis of the requirements for privacy-sensitive ubiquitous computing systems. Chapter 2 outlines our analysis of end-user privacy needs. Chapter 3 details our analysis of application developer needs. Chapter 4 describes our analysis of pitfalls in user interfaces for managing privacy.

The third part looks at the design, implementation, and evaluation of the Confab toolkit. In Chapter 5, we describe the system architecture, data model, and programming model of Confab, as well as specific extensions for location privacy. In Chapter 6, we present an evaluation of Confab through building three applications and user studies of those applications.

We wrap up with a description of related work in Chapter 7, an outline of future work in Chapter 8, and a conclusion in Chapter 9. Supplemental materials are contained in the appendices.

Part II

Requirements Analysis for Privacy-Sensitive Ubiquitous Computing

2 End-User Privacy Needs for Ubiquitous Computing

The primary metric of success for any toolkit is if it can be used to create a useful and non-trivial subset of the full design space of applications in a manner that is faster, is higher quality, or has more useful features than without it. In this and the next two chapters, Chapters 3 and 4, we map out the requirements of this design space, and then continue in Chapter 5 with a description of the Confab architecture and how it makes it easier and faster for developers to create high-quality applications in this design space.

This chapter looks specifically at the end-user privacy needs for ubiquitous computing. As noted in the introduction, our focus here is primarily on helping end-users manage their personal privacy. This is in contrast to other aspects of privacy, such as enterprise support for managing privacy or government policy. Here, we first summarize the end-user privacy needs. Next, we describe the sources used to gather those needs, including research papers, surveys, and interviews, and then continue with a detailed description of the end-user privacy needs.

2.1 Summary of End-User Privacy Needs

Briefly, the privacy needs we gathered are as follows:

- Applications need a *clear value proposition* that makes it obvious what benefits are offered and what personal information is needed to provide those benefits
- People want *simple and appropriate control and feedback* about who can see what information about them
- People expressed a strong desire for *plausible deniability*
- There should be *limited retention of data,* to minimize the risk of extensive data mining and accidental disclosures
- Systems should provide *decentralized control*, giving end-users the greatest amount of choice over how their information is used[3]
- There should be *special exceptions for emergencies*

2.2 Overview of Sources Used

The end-user needs were gathered through a wide variety of sources, each of which make different assumptions and thus have different insights on how to design privacy-sensitive systems. The advantage of using multiple sources is that it provides a broader view of privacy, as well as compensates for the weaknesses of any individual approach. It should be noted that the emphasis here is on privacy needs

[3] It should be noted that there are also several disadvantages to this decentralized approach, namely that data must be periodically updated, that it requires smarter clients with more processing and storage capabilities, and places the burden of system administration on end-users. See subection 8.2.4 for more details.

from the end-user's perspective. As such, we do not use sources describing system architectures, for example GeoPriv [42] or Asymmetric Information Flows [84]. Instead, these are discussed in the next chapter, which looks at application developer needs for privacy-sensitive ubicomp systems.

	Value Proposition	Control and Feedback	Plausible Deniability	Limited Data Retention	Decentralized Control	Exceptions for Emergencies
Theoretical Frameworks on Ubicomp Privacy						
Bellotti and Sellen's Framework for Privacy in Multimedia Spaces [21]		x				
Adams' Framework for Privacy in Multimedia Spaces [7]	x	x				
Palen and Dourish's Boundary Negotiation and Genres of Disclosure [116]		x				
Fair information practices [149] and Langheinrich's extrapolations [93]		x		x	x	
Grudin & Horvitz's Pessimistic, Optimistic, and Mixed Modes [69]		x				x
Desituating Action: Digital Representation of Context [67]	x			x		
Proposed and Existing Data Protection Laws						
Location Privacy Protection Act of 2001 [47]		x		x		
Wireless Privacy Protection Act of 2003 [4]		x				
European Union Directive on Data Protection [51]		x		x		
Published Descriptions of Experiences with Ubicomp Technologies						
Why People Do and Don't Wear Active Badges: A Case Study [71]	x					
Privacy Interfaces for Collaboration [30]	x	x				
Casablanca: Designing Social Communication Devices for the Home [75]		x	x			
Privacy and Security in the Location-enhanced World Wide Web [76]		x			x	
User Needs for Location-aware Mobile Services [86]		x				
Location-based services for mobile telephony [17]					x	
Firsthand Descriptions of Ubicomp Technologies						
Nurse message board on locator systems [1, 12-14]	x					
Analysis of freeform comments from a survey on ubicomp privacy [96]		x	x			
Scenario based interviews of location-based services	x		x			x

Table 2-1. This table lists the various sources used in the analysis of end-user needs for privacy in ubicomp environments, and which of the end-user needs for privacy they influenced.

These sources can be roughly organized into four groups (see Table 2-1): theoretical frameworks on ubicomp privacy, proposed and existing data protection laws, published descriptions of experiences with ubicomp, and firsthand descriptions of experiences and desired privacy preferences with ubicomp technologies.

2.2.1 Theoretical Frameworks on Privacy

In this subsection, we describe several theoretical frameworks on privacy that helped influence our end-user requirements for ubicomp privacy.

Bellotti and Sellen's Framework for Privacy in Multimedia Spaces • Bellotti and Sellen argue that certain designs may be conducive to unethical uses as well as inadvertent intrusions. They argue that proper feedback and control mechanisms can help mitigate or avoid potentially privacy-intrusive features [21]. Using a multimedia ubicomp environment as a case study, they describe a framework for designing appropriate feedback and control mechanisms, looking at issues surrounding capture, that is what kinds of information are being collected; construction, that is how the information is represented and stored; accessibility, that is who has access to the information; and purpose, which is how the information is used.

Bellotti and Sellen's framework influenced the end-user privacy needs by stressing the importance of simple and appropriate forms of control and feedback for end-users.

Adams' Framework for Privacy in Multimedia Spaces • Adams looks at perceived infringements of privacy in multimedia communication systems [7]. Through several user evaluations, she identified three factors that influenced people's perceptions of these systems: information sensitivity, how private a user considered a piece of information; information receiver, who the person receiving the information was; and information usage, how the information is used.

Adams' framework influenced the end-user privacy needs in this work by stressing the importance of the value proposition for end-users (described by Adams as cost / benefit and information usage), as well as simple and appropriate forms of control and feedback for end-users.

Palen and Dourish's Boundary Negotiation and Genres of Disclosure • Palen and Dourish argue that privacy is not simply a problem of setting rules and enforcing them, but rather an ongoing and organic process of negotiating boundaries of disclosure, identity, and time. They also suggest *genres of disclosure* for managing interpersonal privacy, which are "socially-constructed patterns of privacy management," as a sort of design pattern approach to support the development of privacy-sensitive applications [116]. Examples might include creating and managing accounts at shopping Web sites, taking appropriate photographs at social events, exchanging contact information with a new acquaintance, and the kinds of information one reveals to strangers. A person fulfills a role under a genre of disclosure through her performance of her expected role in that genre, and the degree to which a system does *not* align with that genre is the degree to which it fails to support the user's and the genre's privacy regulation process.

Although there are many lessons from this work, the two most salient here are that strict rule-based user interfaces may not always match peoples' needs in managing their privacy, and that social, organizational, and institutional settings are important factors to consider when designing and deploying ubicomp technologies.

Fair Information Practices and Langheinrich's Extrapolation to Ubicomp • The fair information practices are a set of guidelines to help large organizations, such as corporations and governments, manage people's personal information in a responsible manner [149]. They include concepts such as notice, choice, security, and recourse. 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 [93].

The fair information practices, in particular Langheinrich's extrapolation of them to ubicomp, influenced the end-user privacy needs by describing the strong need for control and feedback (especially notice and consent), as well as limited data retention. The idea that personal information should be collected only for express purposes, and that people should be able to access and amend their personal information, helped lead to the idea of emphasizing locality and decentralized control in the implementation of the Confab toolkit.

It should be noted, however, that while the fair information practices have been extremely influential on the field of information privacy and on this work as well, they are intended more for large organizations and do not necessarily translate well for interpersonal relationships between co-workers, friends, and family.

Grudin and Horvitz's Description of Pessimistic, Optimistic, and Mixed Modes • In a workshop position paper, Grudin and Horvitz [69] observed that control and feedback mechanisms for privacy can be generally classified as one of three types: pessimistic, mixed, and optimistic. Their observations complement previous work in which we described three general strategies for managing privacy: prevention, avoidance, and detection [84].

With pessimistic sharing, the goal is to *prevent* privacy intrusions from taking place. In this scheme, users must predict in advance who might want to use their personal information and then set the access privileges accordingly. The problem here is that it can be difficult to predict in advance what permissions are needed, as well as updating those permissions as the situation changes.

With mixed-initiative sharing, the goal is to provide useful information to let end-users make better choices, helping them *avoid* privacy intrusions. An example is choosing whether or not to answer a phone call given the identity of the caller.

People are interrupted when an access request occurs, and can make a decision then and there as to whether or not they wish to share information.

With optimistic sharing, the goal is to help end-users *detect* privacy intrusions and then fix them afterwards. For example, the owner could revoke access privileges after the fact, and possibly rely on other external mechanisms as well, such as firing them. Optimistic sharing is useful in cases where openness and availability are more important than complete enforcement of access. Optimistic sharing is also easier to use, since it is difficult for people to predict all of the possible usage scenarios they might find themselves in, and thus all of the necessary permissions. Furthermore, optimistic sharing provides a level of social translucency that is likely to prevent many kinds of abuses in cases where the parties have an ongoing relationship. For example, Alice is less likely to repeatedly query Bob's location if she knows that Bob can see each of her requests.

It should be noted that most applications will have a mixture of these mechanisms. For example, AT&T Find Friends [15] provides strict access control to restrict who can view one's location information (pessimistic sharing). Authorized users can make as many queries on a person's location as desired, with notifications providing enough social visibility to prevent abuses (optimistic sharing).

Grudin and Horvitz's ideas have influenced the end-user privacy needs by emphasizing that pessimistic, optimistic, and mixed modes can be used to develop appropriate forms of control and feedback. This insight also influenced the design of the access notification user interface in the Confab toolkit, as described in Chapter 5.

Desituating Action: Digital Representation of Context • In an insightful philosophical essay, Grudin points out several issues that must be dealt with when developing context-aware systems [67]. He notes that privacy is a relatively new concept, and that “when people see benefits that outweigh risks, they voluntarily adjust their comfort levels regarding privacy”, citing surveillance cameras as a prime example. However, he also notes that a more fundamental problem is that technology is transforming what it means to be situated. He writes, “We are losing control and knowledge of the consequences of our actions, because if what we do is represented digitally, it can appear anywhere and at any time in the future. We no longer control access to anything we disclose.”

Grudin’s essay influenced the end-user privacy needs with his observation that the value proposition is a major factor affecting how people perceive their own privacy. Grudin also emphasizes the potential dangers of personal data gathered in the past affecting a person in the future, leading to the need for limited data retention.

2.2.2 Proposed and Existing Data Protection Laws

In this subsection, we describe several proposed and existing data protection laws that helped influence our end-user requirements for ubicomp privacy.

Location Privacy Protection Act of 2001 • Although not enacted as law, this proposed bill outlines why the United States government has a substantial interest in protecting individuals' location information. It outlines many potential risks, noting that an adversary could use knowledge of another's location information to "commit fraud, to harass consumers with unwanted messages, to draw embarrassing or inaccurate inferences about them, or to discriminate against them", and that "collection or retention of unnecessary location information magnifies the risk of its misuse or improper disclosure" [47].

The Location Privacy Protection Act of 2001 also outlines several rules that would govern how wireless services, including location-based services and applications, could use an individual's location information. These rules include notice, consent to specific uses as well as disclosures to third parties, purpose, security, and neutrality with respect to the technology to promote fair competition.

The proposed Location Privacy Protection Act of 2001 influenced the end-user privacy needs by emphasizing many different risks involved with location-based technologies, as well as stressing the risks of data retention. It also influenced the control and feedback mechanisms of Confab, with respect to the design of the service descriptions describing what personal information is needed from an end-user to provide a service, as described in Chapter 5.

Wireless Privacy Protection Act of 2003 • This amendment to the Communications Act of 1934 specifies several provisions requiring express consent from users in order to use wireless location and crash information, including "(A) a description of the specific types of information that is collected by the carrier; (B) how the carrier uses such information; and (C) what information may be shared or sold to other companies and third parties." While supporting telephony carriers is not directly within the scope of this work, the ideas embodied by this act helped influence the design of the control and feedback mechanisms of the service descriptions (described in chapter 5).

European Union Directive on Data Protection • The European Union Directive on Data Protection [51] is the most comprehensive of set of data privacy laws currently in existence. In many respects, this directive closely follows the fair information practices described above, and includes several information privacy principles such as data quality (e.g., data is collected for specified purposes only, is not collected excessively, is accurate), legitimate processing (e.g., consent, notification of purpose and sharing, etc), adequate security, and so on.

Many of the information privacy principles embodied by the European Union Directive are well within the state of the art, and thus are not within the scope of this research. However, some of these principles did help influence the development of this work. The notion of data quality, that data should be kept no longer than necessary to fulfill the stated purpose, helped lead to the idea of limited data retention. Other principles relating to notice and consent also influenced the design of the control and feedback mechanisms described in Chapter 5.

2.2.3 Published Descriptions of Experiences with Ubicomp Technologies

In this subsection, we describe several published descriptions of experiences with ubicomp technologies that influenced our end-user requirements for ubicomp privacy.

Why People Do and Don't Wear Active Badges: A Case Study • Harper provides a multifaceted analysis of why some people involved in the initial PARCTab system chose to wear Active Badges while others did not [71]. His work was based on ethnographically informed interviews with 44 people and interpreted through a sociological lens.

Harper uncovered several factors contributing to acceptance or rejection of Active Badges, the most relevant of which was that wearing or not wearing a badge was implicitly seen as a social act representing membership. The group developing applications for the Active Badges saw themselves as stakeholders in the systems that they were developing, and thus did not have many problems wearing the badges. In contrast, this same group was seen as having a techno-centric worldview that ignored issues of utility and privacy, particularly by members of another research group that generally rejected the badges. So while privacy was an important issue, it was also couched in the social and organizational dynamics of the research lab.

Harper's work contributed to the end-user needs for privacy by indirectly noting that the group that did see value in the Active Badges was more likely to wear them than the group that did not see any value. Harper's work is also useful in bringing to light many of the non-technical issues involved in the success or failure of ubiquitous computing technologies.

Privacy Interfaces for Collaboration • In this technical report, Cadiz and Gupta describe the results of two lab studies, seven people each, aimed at understanding people's decision-making processes and concerns when sharing personal information with other individuals, for example with friends, family, and co-workers (as opposed to businesses and governments) [30]. The user interface they developed was based on a spreadsheet metaphor, with types of information as rows and people as columns. Participants in the study were asked to specify how comfortable they would be sharing a piece of information with a specific individual. The user interface also hypothetically included notifications on access as well as time limits on when people would have access, for example for the next five days.

Cadiz and Gupta observed that people often asked four questions when making decisions, including whether or not a requestor already has this information, if the requestor needs to know this information, if it matters if the requestor has this information, and if the requestor is trustworthy.

Cadiz and Gupta influenced the end-user privacy needs of this work by drawing out several factors contributing to the value proposition when sharing with other people, as well as suggesting that notifications and time may be a useful form of control and feedback.

Casablanca: Designing Social Communication Devices for the Home • Hindus et al. examined social communication devices for the home, developing and evaluating several device prototypes that could make it easier for friends and family to have a lightweight awareness of each other [75]. Two key lessons from this work are that these kinds of social communication devices should be designed such that people have control over the timing and type of interaction, and can fulfill existing social obligations without adding new ones, an example of a compelling value proposition for friends and families. This work also suggests that control and feedback mechanisms should be very simple and designed to be a natural part of existing interactions (for example, a presence lamp that automatically lets friends know you are home just by turning on the light).

Privacy and Security in the Location-enhanced World Wide Web • In previous work, we analyzed the privacy risks of the Place Lab system, a bootstrapping effort for enabling ubiquitous location-enhanced systems using WiFi [76]. Place Lab is described in greater detail in Chapter 5.

We also discussed three stumbling blocks faced by the PARCTab system. First, PARCTab used a centralized server to hold location data. While this architecture made it easier to create certain kinds of applications, it meant that sensitive data was stored on a computer that end-users had little practical control over. Even though a visible effort was made to create written privacy policies, users still had the perception that if the research team managing the system changed their policies, or if upper-level managers wanted to examine the data, there was little they could do about it. In addition, centralized servers are attractive targets for computer security attacks.[4]

Second, there was no control over the level of location information disclosed. By design, PARCTab base stations continuously forwarded location information to higher level processes. Even without running applications, the device's location was known because it beaconed a data packet for this purpose. The system was "all or nothing":

[4] In many respects, this issue of centralization has been the main sticking point for E911 systems. The primary concern for E911 is that the government or phone companies could surreptitiously track individuals without their knowledge or consent.

users did not have any granular control over the degree of information sent (it specified location by room) or whether that information was shared with others. There were no provisions for ambiguity or for tailoring the level of disclosure to suit individual preferences.

Third, there was no disclosure over what information was revealed to third parties. A stranger could monitor a user's location by making repeated queries about the user's location without that user knowing.

The analysis of the PARCTab system led to the realization of the risks posed by centralized systems, leading to the end-user need of decentralized control. It also led to the need for simple and appropriate control and feedback. This analysis also introduced the initial mock up of a user interface for managing location information which later formed the basis for the Place Bar, as described in Chapter 5.

User Needs for Location-aware Mobile Services • To understand issues surrounding utility and deployment of location-based services, Kaasinen conducted a series of studies in Finland [86]. This included 13 group interviews with a total of 55 people, which looked at peoples' reactions to scenarios such as location-aware advertising, a visit to an exhibition, different holiday and working trips, meeting friends in the evening, going to work, and shopping. This also included a user study where end-users used location-based services to accomplish specific tasks, and expert evaluations of commercial applications already in use.

Interestingly, Kaasinen found that in the group interviews, people had privacy concerns with location-tracking technologies, but less so with the actual applications using that information. She observed that either most of the interviewees did not realize that they could be located while using a service, or that they had a great deal of faith in their telecom operators. As an example of the latter, one interviewee said, “The telecom operators will guard that kind of information. They already have all kinds of information about me but do not distribute it around.” Many interviewees also believed that there would be regulations and legislation to protect people using such services, though some privacy concerns did arise during the user evaluation portions of the study.

Kaasinen brings to light many social and cultural expectations with respect to location-based services. For example, she conducted her studies in Finland, where people generally have a greater level of trust in companies and governments than in the United States. Kaasinen also observes that a service provider betraying a user’s trust in a system can cause serious long-term economic harm to that service provider. For this reason, she advocates that location-aware services should inform the users of what kind of data is collected, how is it used and who has access to it. Applications should also be designed such that users can flexibly control the release of private information, and remain anonymous if desired. These control and feedback recommendations influenced the design of the access notification and Place Bar user interface components, as described in Chapter 5.

In a 5-day diary study of 16 participants, Barkhuus and Dey found that location-tracking systems, ones that rely on tracking of peoples' location by other parties such as mobile telephony service providers, generated more concerns than position-aware systems, ones that provide services based on the device's own knowledge of its position. Participants were given four descriptions of hypothetical location-based services, and asked to keep track of how often each would have been useful to them during their daily activities. Overall, participants had some privacy concerns with the two location-tracking applications, a lunch service where a retailer could push ads to users when near a restaurant and a notification service that could alert users when friends are within a certain distance, but far fewer concerns with the two position-aware systems, two applications that could turn off the ringer on a cell phone when in a private setting (such as in a meeting or in class) or in a public setting (such as in a movie theater).

Though it is only preliminary evidence, this study suggests that applications that provide local control over the usage of personal information are considered more useful and less intrusive than those applications that share such information. Strategically, it suggests that decentralized control is a useful approach, and that application developers should consider deploying applications that benefit individuals first, to demonstrate the utility of location-based services, before deploying applications that require those individuals to share their personal information.

2.2.4 Firsthand Descriptions of Ubicomp Technologies

In this subsection, we describe several firsthand descriptions of experiences with ubicomp technologies that influenced our end-user requirements for ubicomp privacy.

Message Board Postings from Nurses • We examined several postings on a message board devoted to nurses [1], looking for comments and reactions by nurses on the use of locator systems in their hospitals. These locator systems are essentially Active Badges that let any member of the hospital staff see where a specific nurse currently is. We examined three different threads [12-14], which had a total of 35 nurses participating across 57 posts, as shown in Table 2-2. This analysis was particularly useful as it represents freeform thoughts on the long-term use of a ubicomp system.

A primary insight from this analysis is that the value proposition was an important part of how the nurses perceived the system. In cases where the value proposition was clear to the nurses using it, and where management respected the nurses, the system was accepted. In cases where the value proposition was not clear or was seen as not directly helping the nurses, the system tended to exacerbate existing tensions between the staff and management.

Thread Title	Start Date	#Posts	#Nurses
Nurse Tracking Devices: Whats Your Opinion?	Jun 26 2001	17	10
New Restroom protocol per management....	April 19 2002	13	7
New call lights	July 25 2002	27	20
Total		57	35[5]

Table 2-2. Overview of the threads on the use of locator systems in the nurse message board [12-14].

Freeform Comments from a Survey on Ubicomp Privacy Preferences • We also performed an extended analysis of freeform comments on a previously conducted survey of 130 people on ubicomp privacy preferences [96]. This survey asked several questions about what kinds of personal information participants would be willing to disclose in hypothetical situations, as well as what factors would contribute to those disclosures. The freeform comments from this survey are presented in Appendix C (courtesy of Scott Lederer).

The primary user interface metaphor used in the survey was the notion of a "face." The basic idea was derived from Goffman's observations of how we present ourselves in everyday life [61]. Goffman's key insight is that we present and maintain different personas, or different faces, to different people in different

[5] There were two nurses that posted in two of the threads, so the total adds to 35 rather than 37.

situations, in accordance with our perceived roles in those situations. For example, a doctor might present a professional persona while working in the hospital, but might be far more casual and open while at home, presenting a different aspect of her persona. The relevance here with respect to privacy is that what we are willing to disclose to others is strongly influenced by our expected role in a given situation, which cannot always be easily captured or modeled with existing computer systems.[6]

We examined the freeform comments from the survey responders. It should be noted that the responders included a large number of engineering students from UC Berkeley and was also self-reported. However, we are concerned here less with statistically significant results, and more on common themes expressed by responders. The freeform comments also proved useful in providing insights into how people described their privacy preferences in natural language, specifically that with respect to personal interactions, people often described access in terms of who and when. For example, one survey responder wrote a comment representative of several responders: “during the work day, or after-hours during crunch time, I’d want my boss/coworkers to find m[e] - after hours I’d rather be more anonymous”.

Many responders also stated a need for plausible deniability, especially that white lies should be believable and should not leak information. For example, one

[6] This survey formed the basis for a user interface for managing privacy in ubicomp environments. However, it turned out to be an extremely difficult concept to implement. In fact, the researcher that conducted this survey and created the aforementioned user interface considers his implementation a failure [94]. Together, we did a failure analysis of that user interface, which led to the pitfalls in user interfaces for privacy described in chapter 5.

survey responder wrote "changing the amount of information revealed under different circumstances is information itself. if [you] don't reveal to your friends where you are exactly does that mean you are on a date? saavy and smart people will be able to extrapolate information easily." These insights strongly influenced the design of the access notification user interface described in Chapter 5.

Scenario-based Interviews on Location-Enhanced Applications • We also conducted scenario-based interviews on location-enhanced applications with twenty people of various ages and computer expertise living in the San Francisco Bay Area. In summary, there were 9 males and 11 females; 10 were working professionals and 10 were students; 5 people who considered themselves experts with computers, 14 intermediate, and 1 novice; and 14 of them owning cell phones, 15 of them have used instant messaging, and 6 have used GPS or some other kind of location technology. The full demographics of our participants is described in Table 2-3.

In each interview, we described five different location-enhanced applications to each participant, as described in Appendix A. These applications included:

- a find friends system that would let you query your friends for their current location (and vice versa)
- an active campus map that displayed the real-time location of your friends
- a never-get-lost system that could bring up a map showing where you currently were, your destination, and nearby points of interest

- a mobile e-commerce system that provided physical searches (e.g., “show me all shoes size 9 in this store”), an option for personalized results, and location-specific advertisements
- an emergency response system that used a trusted third party to store your location information in case emergency responders needed it

ID	Age	Gender	Computer Skill	Cell	IM	GPS	Profession
1	26-30	M	Expert	Y	Y	N	College Student (CS)
2	21-25	F	Intermediate	Y	Y	N	College Student (Bio)
3	16-20	F	Intermediate	Y	Y	N	College Student (Psych)
4	21-25	F	Intermediate	Y	Y	Y	College Student (Bio)
5	21-25	F	Intermediate	Y	Y	Y	College Student (Comp Lit/Playwriting)
6	21-25	M	Intermediate	N	Y	N	College Student (EECS)
7	21-25	M	Expert	N	Y	N	College Student (CS)
8	51+	M	Expert	N	Y	N	Engineer, Software
9	21-25	F	Intermediate	Y	Y	N	College Student (EECS)
10	21-25	F	Intermediate	Y	Y	Y	Researcher
11	26-30	F	Intermediate	Y	Y	N	Graphic Designer
12	51+	F	Novice	N	N	N	Registered Nurse
13	51+	M	Intermediate	Y	N	Y	Lawyer
14	51+	M	Intermediate	N	N	N	Scientist
15	51+	M	Intermediate	Y	N	N	CEO
16	46-50	F	Intermediate	Y	N	N	Accountant
17	21-25	F	Intermediate	Y	Y	Y	College Student (Math/Economics)
18	16-20	F	Intermediate	N	Y	N	High School Student
19	51+	M	Expert	Y	Y	Y	Systems Engineer
20	21-25	M	Expert	Y	Y	N	Free Lance Web Designer

Table 2-3. Demographics of interviewees. Ages were grouped into 5-year ranges, for example 21-25 and 26-30. Column “Computer Skill” was a self-reported indication of whether the interviewee considered themselves a novice, intermediate, or expert with computers. Column “Cell” indicates whether they own a cell phone or not. Column “IM” indicates whether they have used IM before. Column “GPS” indicates whether they have used any electronic navigation device before. All interviewees resided in the San Francisco Bay Area.

We also took care not to mention the word “privacy” to interviewees unless they did first. Our interest here was in how they judged the value of each application, as well as who they were willing to share information with and under what conditions, how they thought the application worked, and what concerns they had.

Each interview lasted 45-60 minutes, was conducted at a place of the interviewee’s choosing, and was recorded using a digital voice recorder. Each interview concluded with a short debriefing and closing comments from participants. Specific quotes and concerns from the interviewees are detailed in the following section.

One weakness of scenario-based interviews is that it asks people to place themselves in hypothetical situations to elicit what their attitudes are, which might not be the same as their actual behaviors in that situation. However, we believe that this approach still yields useful information about how location-enhanced systems should be designed, as it represents peoples’ first impressions of a system, as well as some concerns that they may have. It should be noted, though, that application designs should not rely on interviews exclusively, but rather should be used in conjunction with other methods as part of an iterative user-centered process for understanding and designing applications for end-users.

2.3 Discussion of End-User Privacy Needs

From the sources described above, we have drawn six major themes (see Table 2-4), which we describe below in more detail.

	End-user requirements for Ubicomp Privacy
•	Clear value proposition
•	Simple and appropriate control and feedback
•	Plausible deniability
•	Limited retention of data
•	Decentralized control
•	Special exceptions for emergencies

Table 2-4. Summary of end-user requirements for ubicomp privacy.

Clear value proposition • Applications need an upfront value proposition that makes it immediately clear to end-users what benefits are offered and what personal information must be shared to obtain those benefits. Without a strong value proposition, end-users may feel that they have no compelling reason to share information (or even feel resentful if compelled to do so), as it exposes them to risk without any benefit.

One example of this can be seen in the nurses' comments on locator systems. Interestingly, the comments 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 Radio Frequency Identification (RFID) systems in the United States and in England (see for example [19]). From a retailer’s perspective, RFIDs are beneficial because they can be used for tracking inventory, maintaining steady supply chains, and cutting costs. However, from a customer’s perspective, RFIDs are potentially harmful, because they expose customers to the risk of surreptitious tracking without any salient benefit to them at all. It is not surprising that people would have serious privacy concerns here.

In many ways, the issue of value proposition can be considered a variation of Grudin's law [68], 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 in proportion to the perceived risks, then the technology is likely to fail.

Simple and appropriate control and feedback • People want simple control over and feedback about who can see what information about them.

For example, the PARCTab system provided no control about what information was being revealed to others [76, 129]. By design, PARCTab base stations continuously forwarded location information to higher level processes. Even without running applications, the device's location was known because it beaconed a data packet for this purpose. The system was "all or nothing": users did not have any granular control over the degree of information sent (it specified location by room) or whether that information was shared with others. There were no provisions for ambiguity or for tailoring the level of disclosure to suit individual preferences.

The PARCTab system also provided no feedback about what information was revealed to others. There were serious concerns that a co-worker or boss could monitor a user's location by making repeated queries about the user's location without that user ever knowing.

This lack of control and feedback often led people to *suspect* that others were monitoring them, regardless of whether it was actually happening, and was very likely a major factor contributing to the hostility towards the initial work in ubiquitous computing. This is also a perfect example of how ubiquitous computing can unintentionally (or perhaps intentionally) lead to what Bentham termed the Panopticon [22]. First described in the early 19th century and later used as a metaphor for the monitoring and control of individuals by philosopher Michel Foucault [56], the Panopticon was a prison physically designed in such a way so that guards could always see prisoners while the guards themselves remain unseen. The mere threat that a prisoner might currently be under observation would consequently lead him to act only in an "appropriate" manner. One could easily imagine ubicomp technologies being used in a similar manner, to ensure that people only go to "appropriate" places or engage only in "appropriate" activities. This has, in fact, already started to happen. For example, Tennessee has started to use GPS ankle bracelets to monitor parolees [141]. Some car rental companies have used GPS to monitor speeding [97] and to ensure that cars are driven only in pre-specified locations [90]. GTX Corp is selling a GPS-enabled shoes that lets parents monitor children, notifying them when any of several parameters are broken [88]. Some mass transit systems (for example, the BART system in San Francisco) have obvious cameras angled at the passengers, which may or may not actually be recording. Again, the main point here is how a system is actually designed and deployed can make people feel like they are being monitored and consequently control their behavior, regardless of whether they are actually being monitored or not.

There are also concerns about *continuous* versus *discrete* flows of information. Many of our interviewees said they would be comfortable with co-workers getting snapshots of their current location, but would be less comfortable continuously sharing their location information, as that could be used to monitor them. For example, with respect to the find friend application, interviewee #8 said, “I am sensitive to having [to] disclose my information so that someone could find me. I wouldn’t want something to constantly profile me.” Interviewee #16 echoed similar concerns, saying, “Too much invasion of privacy. I don’t want to be watched. I don’t want to be visible to other people, even your friends. It’s too much going on.”

Interestingly, several of the interviewees preferred that their friends call them on their mobile phone rather than using the Find Friend application. For example, interviewee #2 said, “If I want to be found, then I want to be found. I would answer my phone.” Interviewee #20 had similar thoughts, saying, “No, I don’t think I want to share my location. I hide a lot and I don’t want people to find me. They can call me on a cell phone if they want to find me.” We believe they felt this way because they were already familiar with how a cell phone works. Talking on a cell phone makes it clear what information is being disclosed to the other party and gives the speaker the wherewithal to make white lies. The design solution we have adopted, as described in Chapter 5, is to use the optimistic, mixed, and pessimistic modes of sharing as described in Section 2.2.1, allowing those who wish to share as well as those who do not, to do so easily.

In summary, people have many reasons for sharing personal information, but they also want simple control over and feedback about who can see that information.

Plausible Deniability • Many people have also expressed a strong desire for plausible deniability. Our survey and interviews, as well previous work on ubicomp in the home by Hindus et al [75], have suggested a social need to avoid potentially embarrassing situations, undesired intrusions, and unwanted social obligations. For example, it is not uncommon for an individual to answer with a white lie when asked on the phone where they are or what they are doing.

This desire about plausible deniability was mentioned by several participants during our interviews. For example, during the debriefing, interviewee #5 noted that "Nobody can say no in this society. It's easier to be avoidant." Interviewee #14 described his strong desire for an invisible mode, saying "If I don't want to go to the board meeting, mostly because I have relatives home. So I want to be conveniently invisible." Interviewee #7 noted some potential problems with how an invisible mode might be implemented, saying "Say if I was looking for another job, and I don't want my boss to know. Hypothetically if I was cheating on a girlfriend. Invisible mode implies that you're doing something bad and you don't want people to know. The word should be changed to such as *offline*".

With respect to implementation, cell phones are a good example of a system that provides plausible deniability. If a person does not answer a call, it could be for technical reasons—such as being outside of a cell, 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. By default, plausible deniability is maintained without the end-user having to take any special action and without the end-user having to configure anything.

Plausible deniability is also a useful aspect of many instant messaging systems, as observed by Nardi et al [108]. They noted that people could ignore incoming instant messages without offending the sender, because the sender does not know for certain whether the intended recipient is there or not. Consequently, failing to respond is not interpreted as rude or unresponsive. Woodruff and Aoki [152] found similar attitudes with respect to push-to-talk systems[7].

One important design issue with respect to plausible deniability is that information might be accidentally leaked. As noted earlier, if a friend is expecting one level of information (for example, location at the street level) and sees another level (for example, city level) that is only used when out dating, that friend might get suspicious. Another survey responder had a similar observation, noting, “The relationships that I establish with individuals (or companies, in the examples above)

[7] It should be noted, though, that it is more likely that instant messenger systems and push-to-talk systems have this property of plausible deniability by accident rather than being an explicit design criteria.

tend to transcend the activities in which I am engaged; once I choose to trust someone with my information, it’s less important to me to be able to change it moment to moment than to maintain and protect that information consistently.” In short, designs need to make it easy for people to project a desired persona, and thus be careful of the implicit information that is transmitted as well as the explicit information.

Limited Retention of Data • Another concern for users of ubiquitous computing technologies lies with the long-term retention of personal information. The danger here is that retention greatly increases the risk for extensive data mining, accidental disclosures, as well as the unearthing of events far in the past that may be embarrassing or even damaging in a present context.

For this reason, limited data retention is explicitly advocated by many data protection laws such as the European Union Directive [51], by proposed privacy laws such as the Location Privacy Protection Act of 2001 [47], and by privacy frameworks such as the Fair Information Practices [149].

With respect to data retention for location-based advertising, interviewee #1 said that saving “[p]references would be okay, but not information to help them to locate me in the future. I wouldn’t want them to correlate my requests or locations in the future with my past. Only current location. Limit the amount of information they keep.” With respect to the emergency response application, interviewee #8 said, “I guess what I would like to have [is] some control over how long information is kept. I want to know where everything has been for the last hour. I am sensitive to having to disclose my information so that someone could find me. I wouldn’t want something to constantly profile me. Though someone may not be interested in me. I wouldn’t want someone [to] be susceptible.”

Interestingly, limited data retention was noted by none of the survey responders and only by these two interviewees, both of whom had strong technical backgrounds. We believe this is because it is hard to know who is retaining one’s personal data, how they are using that information, and because it is difficult to trace a privacy violation back to the initial cause of data retention. This is likely to change, however, as more and more ubiquitous computing technologies are deployed and as people become more aware of the potential risks involved with these systems. As such, we believe it is an important end-user need that should be addressed sooner rather than later.

Decentralized Control • People are concerned about systems that centralize data. While there are many advantages to centralized architectures, it also means that sensitive data is stored on a computer that end-users have little practical control over [17, 76]. In other words, all someone has to do is flip a switch, and all privacy guarantees are gone.

The PARCTab system [129] faced this issue when it was deployed. While a visible effort was made to create written privacy policies about how location information was used, users still had the perception that if the research team or upper-level managers wanted to examine the data, there was little they could do about it [76].

Similar debates have emerged over the deployment of E911 in the United States. Critics have expressed concerns that location-enhanced phones can be used to push location-based spam advertising or to surreptitiously track individuals on a widespread scale.

It is important to note that decentralized control was not noted by any of the nurses in the nurse message board, nor by any of the survey responders or interviewees. The reason for this is that it is subtle and a relatively low-level implementation issue. However, we believe that this is an important need for end-user privacy, because as discussed in the evaluation in Chapter 6, many people assume that most location-based services (with the exception of those using active badges) are decentralized, regardless of whether it is true or not. It is also an issue that seems to cause consternation among individuals when they discover that their information is being stored centrally without their consent.

Special Exceptions for Emergencies • Lastly, people expressed the desire for special exceptions for emergencies. In crisis situations, safety far outweighs privacy needs. This sentiment was universal across all of our interviewees, though some people expressed concerns about the specific implementation and the possibility for abuse. Trusted proxies are sometimes used to handle these kinds of emergency situations. For example, MedicAlert [2] is a paid service that stores personal medical records and forwards it to emergency responders in the case of medical emergencies. In the interview, we asked participants their thoughts about a few different emergency response applications, including E911 and a trusted proxy that could hold your location information in case it was needed.

Interviewee #7 expressed the strongest concerns, saying, “I don’t see how a government or an organization will not come up with an excuse to use it for another purpose”, and “All these things can be twisted in a way so that they can be used for other purposes.” Interviewee #13 was looking for a balance, saying, “I think it’s an invasion of privacy. I should have the ability to call emergency services, but I don’t want them to know of my whereabouts 24/7. I agree with the idea. If a fire truck drives up to the street and they hit a screen, and they could tell that there are four adults, one is over 70.”

Interviewee #12, a nurse, had an interesting perspective, suggesting that these kinds of applications be used in narrow cases. She said, “If there was a device, [it would be] helpful for [people with] Alzheimer’s disease, because they forget. Then the police can track them. Only certain diseases. Useful for kids… Health-risk patients such as diabetes. And if they have seizure disorder. I think it’s very useful. Especially for elderly. If one has Alzheimer’s, the family would inform the police and they can track the woman down. There’s a lot of people [that] run away. I have seen many in the hospital. Especially in an emergency.”

In general, interviewees agreed that E911 made sense if it transmitted location information only when making the call, and not at any other time.

2.4 Summary

In this chapter, we described several end-user needs for ubicomp privacy. These needs were gathered through a variety of techniques, including scenario-based interviews, surveys, posts from a nurse message board, synthesis of previous research, and examination of some proposed and existing laws on data privacy.

In summary, these end-user needs were having a clear value proposition, simple and appropriate control and feedback, plausible deniability, limited retention of data, decentralized control, and special exceptions for emergencies.

3 Developer Privacy Needs for Ubiquitous Computing

In the previous chapter, we looked at privacy from the end-user perspective, synthesizing a set of end-user needs for ubicomp privacy. In this chapter, we examine privacy from the perspective of application developers, focusing on helping application developers construct programs within this design space.[8]

The application developer needs for Confab were gathered by identifying privacy functions common in several networked and ubicomp applications. We examined several research prototypes and emerging commercial applications, limiting the scope to systems where data starts with the end-user and can optionally be disclosed to others in a limited manner (i.e., personal ubiquitous computing rather than ubiquitous computing for places[9]). We also chose to focus more on location than on other forms of contextual information, since a sizeable number of this type of application is emerging in the market, and thus has a clearer path to widespread use. We were also influenced by the Geopriv working group's requirements for location privacy [42], P3P [40], and our previous work on asymmetric information flows [84].

[8] Parts of this chapter were previously published as [77] in The Second International Conference on Mobile Systems, Applications, and Services (Mobisys 2004)

[9] By *personal ubiquitous computing*, we mean ubicomp systems primarily meant for and surrounding a specific individual, for example mobile and wearable systems. By *ubiquitous computing for places*, we mean ubicomp systems deployed in a specific place, for example a smart room or smart kitchen.

The genres of applications we have examined include messaging systems, such as cell phones, instant messenger, SMS, and messaging within [107] and between homes [75]; guides for exploration and navigation [5, 114]; finders for finding people, places, or things [15, 66]; group awareness displays [45, 66]; augmented-reality games [52, 60]; contextual tagging and retrieval, including personal memory aids [27, 91, 125], associating topical information with places [29, 49, 118, 130]; situational real-time information (such as local weather or traffic); and enhanced safety for individuals and emergency responders [53, 104].

3.1 Application Developer Privacy Needs

From a systems standpoint, there are several basic features that need to be supported, including acquiring context data from a variety of sources, refining and storing that context data, and retrieving and using context data. This last issue, retrieving and using, can be done either through *push transactions* (e.g., you send your location in an E911 call) or *pull transactions* (e.g., a friend requests your location). For each of these types, there is also a need for *continuous* sharing, where personal data is constantly forwarded to another party (e.g., continuously sharing health information with your doctor), as well as for *discrete* disclosures that happen intermittently or one time only. These are basic features that are mostly supported by other systems aiding the development of ubicomp applications (e.g. [45, 129]).

From a privacy standpoint, we have identified five common features that need to be supported (see Table 3-1). We discuss these below.

Application Developer Requirements for Ubicomp Privacy
• Support for optimistic, pessimistic, and mixed-mode applications
• Tagging of personal information
• Mechanisms to control the access, flow, and retention of personal information (quantity)
• Mechanisms to control the precision of personal information disclosed (quality)
• Logging

Table 3-1. Summary of developer requirements for privacy-sensitive ubicomp applications.

Support for Optimistic, Pessimistic, and Mixed-mode applications • The first requirement is support for three basic interaction patterns for privacy-sensitive applications as described by Grudin and Horvitz: pessimistic, optimistic, and mixed mode [69]. Also discussed as an end-user privacy need in Section 2.2.1, here our focus is on supporting application developers in creating these kinds of applications.

As a brief recap, in *pessimistic* applications, end-users set up preferences beforehand, placing strict requirements on when personal information can flow to others. In contrast, *optimistic* applications [121] are designed to allow greater access to personal information but make it easier to detect abuses after the fact with logs and notifications. For example, AT&T mMode’s Find Friends [15] provides a notification each time a friend requests your location. In *mixed-mode* control, end-users are interrupted when someone requests their personal information and must make a decision then and there. An example is choosing whether or not to answer a phone call given the identity of the caller.

Tagging of Personal Information • The second requirement is support for tagging personal information as it flows to others, as described by Geopriv [42] and by Korba and Kenny [89]. Personal information can be marked with preferences about, for example, whether it should be forwarded to others or how long it should be retained. These tags can be thought of as applying Digital Rights Management for privacy purposes, and can be used as a fingerprint to help with tracking and auditing as well.

Mechanisms to Control the Access, Flow, and Retention of Personal Information • The third developer privacy need is mechanisms for controlling the access, flow, and retention of personal information, i.e. the quantity of personal information disclosed. These include restrictions based on identity, location (e.g., only allow inquirers in the same building as me to see my location), and time (e.g., co-workers can see my location between 9AM and 5PM), as well as invisible mode, a common feature in instant messenger clients where no information is disclosed.

Mechanisms to Control the Precision of Personal Information Disclosed • The fourth necessary feature is granular control over the precision of disclosures, i.e. the quality of disclosures. One could choose to disclose one’s location as “123 Main Street” or “Atlanta”, or one’s activity as “writing a paper” or “busy”.

Logging • The fifth common privacy feature is logs, both for clients and servers. On the client side, logs that are summarized in a compact form make it easier for end-users to understand who is accessing what data. On the server side, logs make it easier for service providers to audit their activities to ensure that they are handling their customers’ personal information properly. On both sides, logs also make it possible to apply machine learning techniques to detect unusual access patterns that might indicate abuses of someone’s personal information.

3.2 Summary

In this chapter, we described several application developer needs for constructing privacy-sensitive ubicomp applications. These needs were gathered through an analysis of several ubicomp applications, primarily those using location information.

In summary, these application developer needs were having support for optimistic, pessimistic, and mixed-mode applications; tagging of personal information; mechanisms to control the quantity of information flow; mechanisms to control the quality of information disclosed; and logging.

4 Pitfalls in User Interfaces for Privacy

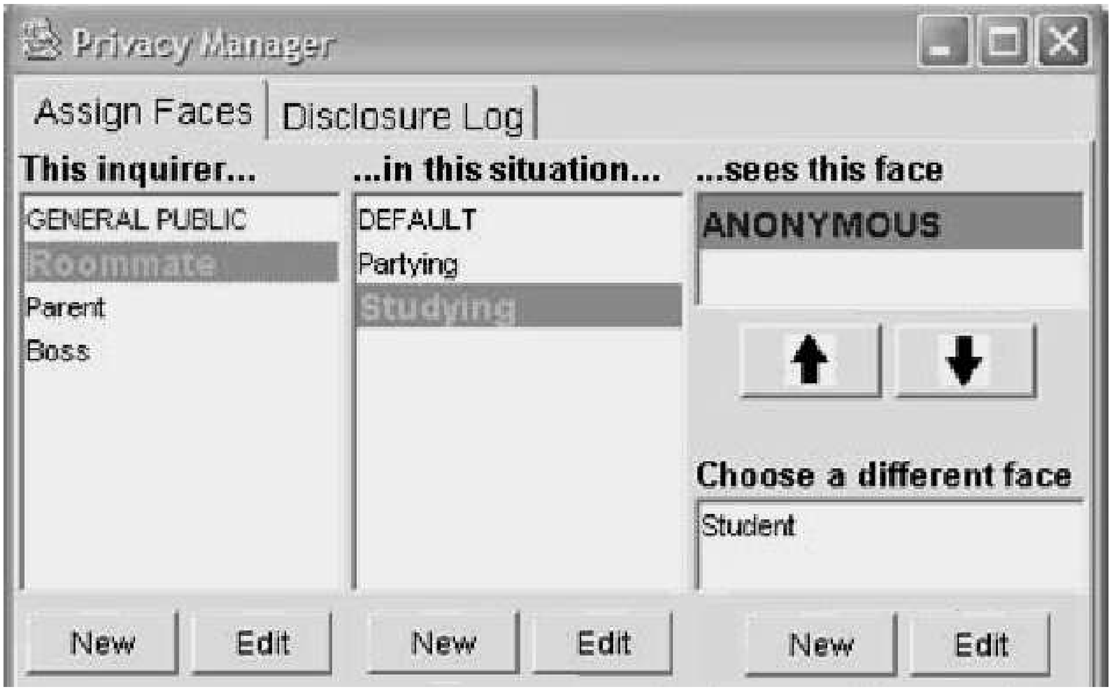
In this chapter, we describe a set of five pitfalls in designing user interfaces for privacy.[10] These pitfalls came about from a failure analysis, jointly conducted with my colleague Scott Lederer, of a user interface he developed for managing personal privacy in ubicomp environments [94], as well as an analysis of 40 other user interfaces for managing privacy. These pitfalls are not a complete guide to creating effective user interfaces for managing privacy, but rather a collection of common design mistakes that on may seem obvious but are still happening. We also look at some ways of avoiding these pitfalls. These pitfalls were used to inform the design of our user interfaces for privacy as described in Chapter 5.

We first provide more background on this work, continue with a summary of the pitfalls, and then proceed into a detailed description of these pitfalls.

4.1 Background

This work came about from a failure analysis of the Faces UI for managing privacy in ubicomp environments developed by a colleague (see Figure 4-1). Full details of this work are described in [94], here we provide a short summary.

[10] Parts of this chapter were previously published as [95] in the journal Personal and Ubiquitous Computing.



This inquirer...

GENERAL PUBLIC
Roommate
Parent
Boss

...in this situation...

DEFAULT
Partying
Studying

...sees this face

ANONYMOUS

Choose a different face

Student

Figure 4-1. The Faces user interface lets people set different disclosure preferences based on inquirer and situation. For example, the current setting shown above is, "if my roommate inquires while I am studying, show my anonymous face," which means no information.

The unifying metaphor of this user interface was based on Goffman's insights on how we present ourselves in everyday life [61]. Goffman observed that we often play different roles in life, and in these roles we present different aspects of our personas, or different faces, to different people in different situations. For example, many people have a professional persona that they project and maintain with colleagues, but a more private one used with family and close friends. The main idea behind the Faces user interface was to make this idea concrete, allowing people to create "faces" that would contain disclosure preferences of who could see what and when, and then set when those faces would be seen by others. For example, "if my roommate inquires while I am partying, show my precise face" (all information), but "if my parents inquire while I am partying, show my vague face" (less information).

The design of this user interface was informed by a series of formative techniques, including surveys, interviews, and low-fidelity prototypes. However, despite this effort, an informal user study of a working Faces prototype showed that people could not successfully set preferences or correctly understand what they were disclosing to others. Furthermore, when asked what kinds of information they were willing to disclose in specific scenarios, it turns out that end-users' stated preferences in natural language often sharply differed from the user interface preferences they had set only minutes before.

Together, we did a failure analysis on the Faces user interface to gain a deeper understanding of why exactly it failed. We examined what kinds of mistakes were made, and also looked if other user interfaces have made these same mistakes as well. We examined over 40 different user interfaces that dealt with privacy, and distilled these mistakes into the pitfalls described below.

4.2 Summary of Pitfalls

We have grouped the pitfalls in user interfaces for privacy into two categories, those that affect users' *understanding* of a system's privacy implications and those that affect their ability to conduct socially meaningful *action* through the system.

UNDERSTANDING

Obscuring potential information flow • Designs should not obscure the nature and extent of a system's *potential* for disclosure. Users can make informed use of a system only when they understand the scope of its privacy implications.

Obscuring actual information flow • Designs should not conceal the actual disclosure of information through a system. Users should understand what information is being disclosed to whom.

ACTION

Emphasizing configuration over action • Designs should not require excessive configuration to manage privacy. They should enable users to practice privacy as a natural consequence of their normal engagement with the system.

Lacking coarse-grained control • Designs should not forgo an obvious, top-level mechanism for halting and resuming disclosure.

Inhibiting established practice • Designs should not inhibit users from transferring established social practice

4.3 Detailed Description of the Pitfalls

In this section, we provide a detailed description of each of the five pitfalls, providing an overview, examples of systems that fall into these pitfalls, as well as examples of systems that avoid them.

4.3.1 Pitfall #1 – Obscuring potential information flow

Systems should make clear the nature and extent of their *potential* for disclosure. Users will have difficulty appropriating a system into their everyday practices if the scope of its privacy implications is unclear. This scope includes the types of information the system conveys, the kinds of observers it conveys to, the media through which it is conveyed, the length of retention, the potential for unintentional disclosure, the presence of third-party observers, and the collection of meta-information like traffic analysis.

Clarifying a system's potential for conveying personal information is vital to users' ability to predict the social consequences of its use. Among the conveyable information types to elucidate are identifiable *personae* (e.g., true names, login names, email addresses, credit card numbers, social security numbers) and monitorable *activities* (broadly, any of the user's interpretable actions and/or the contexts in which they are performed, e.g., locations, purchases, clickstreams, social relations, correspondences, audio/video records).

Privacy-affecting systems tend to involve disclosure both between people and between a person and an organization. Designs should address the potential involvement of each, clarifying if and how primarily interpersonal disclosures (e.g., chat) involve incidental organizational disclosures (e.g., workplace chat monitoring) and, conversely, if and how primarily organizational disclosures (e.g., workplace cameras) involve secondary interpersonal disclosures (e.g., mediaspaces).

“Privacy” is a broad term whose unqualified use as a descriptor can mislead users into thinking a system protects or erodes privacy in ways it does not. Making the scope of a system’s privacy implications clear will help users understand its capabilities and limits. This in turn provides grounding for comprehending the *actual* flow of information through the system, addressed in the next pitfall.

Examples: Falling into the Pitfall

An easy way to obscure a system’s privacy scope is to present its functionality ambiguously. In trying to be a general user interface for managing privacy across any ubicomp system, the Faces system abstracted away the true capabilities of any underlying system. Users could not gauge its potential information flow because it aimed to address *all* information flow. Its scope was impractically broad and effectively incomprehensible.

The Internet control panel in Microsoft Windows has similar problems with ambiguity. This control panel offers ordinal degrees of privacy protection, ranging from Low to High. The functional meaning of this scale is unclear to average users. Furthermore, despite being a component of the operating system's control panel, this mechanism does not control general privacy for general Internet use through the operating system; its scope is limited only to a particular web browser's cookie management heuristics.

Similarly, Anonymizer.com's free anonymizing software can give the impression that all Internet activity is anonymous when the service is active, but in actuality it only affects web browsing, not email, chat, or other services. Instead, a for-pay version covers those services.

Another example is found in Beckwith's report of an eldercare facility that uses worn transponder badges to monitor the locations of residents and staff [20]. Many residents perceived the badge only as a call-button (which it was) but not as a persistent location tracker (which it also was). They did not understand the disclosures it was capable of facilitating.

Similarly, some hospitals use badges to track the location of nurses for efficiency and accountability purposes but neglect to clarify what kinds of information the system conveys. Erroneously thinking the device was also a microphone, one concerned nurse wrote, "They've placed it in the nurses' lounge and kitchen.

Somebody can click it on and listen to the conversation. You don’t need a Big Brother overlooking your shoulder” [123].

A recent example of a privacy-affecting system that has given ambiguous impressions of its privacy implications is Google’s Gmail email system. Gmail’s content-triggered advertisements have inspired public condemnation and legal action over claims of invading users’ privacy [16]. Some critics may believe that Google discloses email content to advertisers—which Gmail’s architecture prohibits—while some may simply protest the commercial exploitation—automated or not—of the content of personal communications. Despite publishing a conspicuous and concise declaration on Gmail’s homepage that “no email content or other personally identifiable information is ever provided to advertisers” [64], the privacy implications of Gmail’s use were unclear to many users when it launched.

Examples: Avoiding the Pitfall

Many web sites that require an email address for creating an account give clear notice on their sign-up forms that they do not share email addresses with third parties or use them for extraneous communication with the user. Clear, concise statements like these help clarify scope and are becoming more common.

Tribe.net is a social networking service that carefully makes clear that members’ information will be made available only to other members within a certain number of degrees of social separation. Of course, this in no way implies that users’ privacy is particularly safeguarded, but it does make explicit the basic scope of potential disclosures, helping the user understand her potential audience.

4.3.2 Pitfall #2 – Obscuring actual information flow

The previous pitfall states that a lack of understanding of what a system in theory can do will make it difficult to put that system into everyday use. This pitfall asserts that a lack of understanding of the actual information flow in a system will similarly make it difficult to use that system. As an example of the difference between the two, with AT&T's Find Friend application [15], the potential information flow is that friends can use the system to check one's location, while the actual information flow is who specifically has checked one's location, such as "Bob saw that you were at the Krispy Kreme at 10:05PM last night".

Exposing the actual information flow in a system is essential because many ubicomp systems are invisible by default. These systems often collect and disseminate personal information without users knowing, thus making it difficult for end-users to understand who is actually seeing what about them. To whatever degree is reasonable, designs should make clear the actual disclosure of information in a way that is obvious and does not overwhelm.

By avoiding both this and the prior pitfall, designs can help end-users understand how their actions are reflected by the system and communicated to others. This can help users understand the consequences of their use of the system thus far and predict the consequences of future use.

Examples: Falling into the Pitfall

Faces conveyed actual information flow through the disclosure log. While this design illuminated the information flow, it is unclear whether postponing notice is optimal. Embedding notice directly into the real-time experience of disclosure might foster a stronger understanding of information flow.

Another example of vague information flow can be seen with web browser support for cookies [106]. Most browsers do not, by default, indicate when a site sets a cookie or what information is disclosed through its use. The prevalence of third-party cookies and web bugs (tiny web page images that facilitate tracking) exacerbates users' ignorance of who is observing their browsing activities.

Muddled information flow can also be seen in the Kazaa P2P file-sharing application, which has been shown to facilitate the concealed disclosure of highly sensitive personal information to unknown parties [63].

Another example is worn locator badges like those described in [20, 71], which generally do not inform their wearers about who is locating them.

Examples: Avoiding the Pitfall

Friedman et al's. redesign of cookie management reveals *what* information is disclosed to *whom*. They extended the Mozilla web browser to provide prominent visual feedback about the real-time placement and characteristics of cookies, thereby showing users what information is being disclosed to what web sites [57].

Instant messaging systems tend to employ a symmetric design that informs the user when someone wants to add him to her contact list, allowing him to do the same. This way he knows who is likely to see his publicized status. Further, his status is typically reflected in the user interface, indicating exactly what others can learn about him by inspecting their buddy lists.

AT&T's mMode Find Friends service, which lets mobile phone users locate other users of the service, informs the user when someone else is locating them. They learn *who* is obtaining *what* information.

4.3.3 Pitfall #3 – Emphasizing configuration over action

Designs should not require excessive configuration to maintain one's personal privacy. Instead, they should enable users to manage their privacy as part of their primary tasks and in the actual context of use.

One problem with configuration is that it requires people to predict in advance what their preferences will be, often in an abstract setting far from the actual context of use. Previous work by Mackay has shown that preferences are often hard to get right, and are often required when people first use a system, at a time when people are least familiar with a system [103].

Another problem with configuration is that, as Palen and Dourish have noted [116], the process through which people maintain their privacy is often an organic and intuitive process rather than one that can be easily defined by rule-based systems. They write, "setting explicit parameters and then requiring people to live by them simply does not work, and yet this is often what information technology requires… Instead, a fine and shifting line between privacy and publicity exists, and is dependent on social context, intention, and the fine-grained coordination between action and the disclosure of that action". However, configuration has become a common interaction design pattern [142], where people are expected to just state upfront what they expect and what they want to make the system work correctly.

Whitten makes a similar observation in the field of security, remarking that security is often a secondary goal rather than a primary goal [151]. In a user study examining usability and security with respect to encrypted email, she notes that people focused on their main goal of sending an email and simply expected security to be included. The mismatch, however, is that existing encryption systems require people to be aware of this implicit goal and then take special actions that are indirect to the main goal to make things work correctly.

Lastly, there is the question of whether or not people will actually go through the effort of configuring a system. For example, a study by Palen showed that most people leave their preferences for group calendars as the default settings [115].

In summary, people generally do not set out to explicitly protect their privacy. Rather, they participate in some activity, with privacy regulation being an embedded component of that activity. Designs should take care not to extract the privacy regulation process from the activity within which it is normally conducted.

Examples: Falling into the Pitfall

Configuration was one of the main stumbling blocks with the Faces user interface. Users had to predict all of the people who might want to request their information, all of the potential situations they might be in, and all of the faces they would want disclosed, before they actually used the system and could understand the implications of use.

Many other systems emphasize explicit configuration for managing privacy, including experimental online identity managers [25, 82], P2P file-sharing software [63], web browsers [106], and email encryption software [151]. In the realm of ubiquitous computing, both our Faces prototype and Bell Labs's Houdini Project [79] require significant configuration efforts prior to and after disclosures.

Examples: Avoiding the Pitfall

Successful solutions might involve some measure of configuration but tend to embed it into the actions necessary to use the system. Web sites like Friendster.com and Tribe.net allow users to regulate information flow by modifying representations of their social networks—a process that is embedded into the very use of these applications.

Dodgeball.com's real-time socio-spatial networking service also directly integrates privacy regulation into the primary use of the system. Dodgeball members advertise their location by sending a brief text message from their mobile device to Dodgeball's server, which then re-sends this message to that member's friends and friends of friends within walking distance. Identifying one's friends to the system does require specific configuration effort, but once done, regulating location privacy is integrated with the very use of the system. Each use actively publicizes one's location; concealing one's location simply involves not using the system.

Ignoring the moral implications, another example involves camera surveillance. When someone is under surveillance, she tends to adjust her behavior to present herself in alignment with the perceived expectations of her ostensible observers [56]. She does not step outside herself to reconfigure her representation. She simply acts, albeit with "appropriate" intuition and/or intention.

Cadiz and Gupta propose a smart card that one could hand to a receptionist to grant him limited access to one's calendar to schedule an appointment; he would hand it back right afterwards. No one would have to fumble with setting permissions. They also suggest extending scheduling systems to automatically grant meeting partners access to the user's location during the minutes leading up to a meeting, so they can infer his arrival time. The action of scheduling a meeting would imply limited approval of location disclosure [30].

4.3.4 Pitfall #4 – Lacking coarse-grained control

Many systems provide a number of flexible, fine-grained controls for managing privacy. The problem, however, is that these systems often make these fine-grained mechanisms the primary form of control while overlooking simpler coarse-grained ones.

While useful, fine-grained controls can make it difficult to understand what the various options are and whether these options are set correctly. From an end-user's perspective, fine-grained controls require a fair amount of effort but results in uncertainty as to whether all of the options were set correctly. This is a common pitfall to fall into because fine-grained control is a common part of computer science. Many application developers are often experts at using computers and desire precise control over every possible aspect of an application, forgetting that this often makes things harder to use and understand for average users.

In the majority of cases, coarse-grained controls offer simpler and clearer conceptual models. For example, many designs offer an obvious, top-level mechanism for halting and resuming disclosure. Users are accustomed to turning a thing off when they want its operation to stop. Often a power button or exit button will do the trick.

It is also easier to reflect the state of a system with coarse-grained controls, providing direct feedback and freeing the user from having to remember whether she set a preference properly. This helps users accommodate the controls and even co-opt them in ways the designer may not have intended. Examples specific to privacy include: setting a door ajar, covering up or repositioning cameras [21, 81], turning off a phone or using its invisible mode rather than navigating its privacy-related options, and removing a worn locator badge.

The main point here is not that systems should not have fine-grained controls for managing privacy, but that coarse-grained controls rather than fine-grained ones should be the primary form of control.

Examples: Falling into the Pitfall

E-commerce web sites typically maintain users' shopping histories. While this informs useful services like personalization and collaborative filtering, there are times when a shopper does not want the item at hand to be included in his actionable history; he effectively wants to shop anonymously during the current session (beyond the private transaction record in the merchant's database). For example, the shopper may not want his personalized shopping environment—which others can see over his shoulder—to reflect this private purchase. In our experiences, we have encountered no web sites that provide a simple mechanism for excluding the current purchase from our profiles.

Similarly, most web browsers still bury their privacy controls under two or three layers of configuration panels [106]. While excessive configuration may itself be a problem (see Pitfall Three), the issue here is that there is typically no top-level control for switching between one's normal cookie policy and a "block all cookies" policy. Third-party applications that elevate cookie control widgets have begun to appear (e.g., GuideScope.com).

Further, wearable locator-badges like those described in [72] and [20] do not have power buttons. One could remove the badge and leave it somewhere else, but simply turning it off would at times be more practical or preferable.

Examples: Avoiding the Pitfall

Systems that expose simple, obvious ways of halting and resuming disclosure include easily coverable cameras [21], mobile phone power buttons, instant messaging systems with invisible modes, the In/Out Board [44], and our Faces prototype.

4.3.5 Pitfall #5 – Inhibiting established practice

People already manage their personal privacy through a range of established and often nuanced practices, and systems should be designed, if not to support these practices, to avoid inhibiting them. One common practice used by people is to tell white lies rather than giving a direct answer that may hurt another person's feelings, such as saying that you are too busy to talk right now. Another related practice is to provide ambiguous answers to questions, such as saying you are "out with friends," rather than saying specifically which friends. These kinds of practices can help provide people with a level of plausible deniability that gives them maneuvering room later on.

The problem, however, is that technical systems are notoriously awkward at supporting these kinds of social nuances [6]. Although people can develop new practices for new technologies, for example adapting to the lack of eye gaze in video conferencing, it can be difficult to predict and design for these practices, and the ones that do emerge generally do not happen as optimally as we might like. Designers will continue to struggle to support emergent practices, but, for now, can at least make sure not to inhibit existing ones.

Examples: Falling into the Pitfall

Some researchers envision context-aware mobile phones that disclose the user's activity to the caller to help explain why their call was not answered [133]. But unless done properly, designs like these can prohibit users from exploiting plausible deniability. There can be value in keeping the caller ignorant of the reason for not answering.

Location-tracking systems like those described in [71] and [20] constrain users' ability to incorporate ambiguity into their location disclosures. Users can only convey their concise location or—when permitted—nothing at all.

Returning to the privacy controversy surrounding Google's email system, one possible reason for people's discomfort with Gmail's content-triggered advertising is its inconsistency with the long-established expectation that the content of one's mail is for the eyes of the sender and the recipient only. With respect to this pitfall, the fact that Gmail discloses no private information to advertisers, third-parties, or Google employees is not the issue. The issue is the plain expectation that mail service providers (electronic or physical) will interpret a correspondence's meta-data (electronic headers or physical envelopes) but never its contents. Many people would express discomfort if the US Postal Service employed robots to open people's mail, scan the contents, reseal the envelopes, and send content-related junk mail to the recipient. Even if no private information ever left each robot, people would react to the violation of an established social expectation, namely, the inviolability—under normal conditions—of decidedly private communications.

Examples: Avoiding the Pitfall

Mobile phones, push-to-talk phones [152], and instant messaging systems [108] let users exploit plausible deniability by not responding to hails and not having to explain why.

Although privacy on the web is a common concern, a basic function of HTML allows users to practice ambiguous disclosure. Forms that let users enter false data facilitate anonymous account creation and service provision.

Tribe.net supports another established practice. It allows users to cooperatively partition their social networks into *tribes*, thereby letting both pre-existing and new groups represent themselves online, situated within the greater networks to which they are connected. In contrast, Friendster.com users each have a single set of friends that cannot be functionally partitioned.

4.4 Summary

In this chapter, we described five common pitfalls in user interface design for privacy-affecting systems. These pitfalls are not a complete guide to creating effective user interfaces for managing privacy, but rather a collection of common design mistakes that on may seem obvious but are still happening. The first two of these pitfalls—*obscuring potential information flow* and *obscuring actual information flow*—look at how people understand a given system. The remaining three pitfalls—*emphasizing configuration over action*, *lacking coarse-grained control*, and *inhibiting established practice*—look at how people can conduct socially meaningful action through the system.

Part III

Design, Implementation, and Evaluation of the Confab Toolkit

5 Confab System Architecture

In this chapter, we describe the design and implementation of the Confab toolkit, which provides an extensible framework for building privacy-sensitive ubicomp applications.[11] The design of Confab was motivated by the analyses and field work described in Chapters 2-4. We start with two example usage scenarios and give a rough description of how Confab supports these. We continue with an overview of Confab's high-level architecture. We then outline the data model, which explains how the data is represented and how it flows between entities. Then, we describe the programming model, which looks at the specifics of how a programmer would develop applications using Confab. We close with a description of extensions for location privacy built within Confab's programming framework.

5.1 Example Usage Scenarios

In this section, we describe two example usage scenarios to help illustrate what kinds of applications we want to support and roughly how they would work within Confab.

[11] Parts of this chapter were previously published as [77] in The Second International Conference on Mobile Systems, Applications, and Services (Mobisys 2004)

Scenario 1 – Find Friend

Alice's workplace has set up a new server that employees can use to share their location information with one another. Employees initially get their location information through beacons describing what room the person is in. They can then choose to share their location information by automatically uploading updates from a personal device (e.g. a cell phone) to the server at the level they desire, for example at the room level, at the floor level, or just "in" or "out". To help allay privacy concerns, the server is also set up to provide notifications to a person whenever their location is queried, and to accept queries only if the requestor is physically in the same building.

Scenario 2 – Mobile Tour Guide

Alice is visiting Boston for the first time and wants to know more about the local area. She already owns a location-enabled device, so all she needs to do is find a service that offers an interactive location-enhanced tour guide and link her device to it. She searches online and finds a service named Bob that offers such tour guides for a number of major cities. She decides to download it and try it out.

When starting the application, Alice discovers that Bob offers three levels of service. If Alice chooses to share her location at the *city level*, Bob can tell her how long the lines are at major venues such as museums, and what calendar events there are. If she shares it at the *neighborhood level*, Bob can also tell her what interesting shops there are and nearby points of interest. If she shares it at the *street level*, Bob can offer Alice all of the features described above, as well as real-time maps and a route finder that can help her navigate Boston. The application also states that Bob will retain her location data for three months, and at the neighborhood level sends updates of her location to Bob every ten minutes when the application is running.

Since this is her first time using the service, and since she has not heard of Bob before, Alice decides to share her location information at the neighborhood level.

5.2 High-Level Architectural Overview

From a high-level perspective, Confab is a hybrid blackboard and dataflow architecture. Personal information is stored in infospaces that are running in computers owned by end-users, with data flowing between computers in a controlled fashion. Below, we provide a brief overview of Confab from two complementary perspectives, decomposing the system architecture into three separate layers as well as describing the dataflow between components in these layers. We also describe the design rationale behind many of the major architectural decisions.

Multiple Layers for Managing Privacy

Roughly speaking, Confab's system architecture can be divided into three orthogonal layers, each of which is responsible for managing and providing privacy protection for different aspects of the flow of personal information (see Table 5-1). These layers include the *physical / sensor layer*, which is responsible for initially capturing personal information; the *infrastructure layer*, which is responsible for storing and processing personal information; and the *presentation layer*, which is responsible for providing user interfaces to give end-users control and feedback over their personal information.

Layer	Responsibility	Examples from Previous Work
Presentation	How information is presented to end-users	P3P [40], Privacy Mirrors [110]
Infrastructure	Where information is stored, how processed	PARCTab System [129], Context Toolkit [45]
Physical / Sensor	How information is captured and gathered	Cricket Location Beacons [122], Active Bats [146]

Table 5-1. Effective ubicomp privacy requires support from three different layers, each of which manages and provides privacy protection for different aspects of the flow of personal information. Previous work has only addressed at most one of the layers. Confab provides support at all three of these layers to facilitate the construction of privacy-sensitive applications.

We argue that effective ubicomp privacy requires support at all of these layers. A system might provide good user interfaces for helping people understand and manage the flow of personal information (presentation), but provide little or no real control over how the information is initially captured (physical / sensor) or how it is processed (infrastructure). The danger here is that one’s privacy can be compromised without that user ever knowing. This is essentially the risk behind centralized systems as described in Section 2.3, in that administrators of those systems can accidentally disclose or maliciously access one’s personal information.

On the other hand, there have been many approaches that provide privacy at the physical / sensor layer. For example, Cricket [122] and Active Bats [146] are both decentralized approaches that use beacons to send out information that lets people determine where they are[12]. However, there is little or no support for helping application developers process that information securely or present it to end-users.

Again, effective ubicomp privacy requires support from the physical / sensor, the infrastructure, and the presentation layers. The problem, however, is that previous work has only addressed one of these layers (see Table 5-1). For example, P3P [40] and Privacy Mirrors [110] look at methods for communicating privacy policies in machine readable formats and providing visibility of tracking mechanisms to end-users respectively, but do not provide any support for capturing or processing personal information in a privacy-sensitive manner. Similarly, while the PARCTab system [129] and the Context Toolkit [45] provide support for processing and storing personal information, they do not provide any features for managing end-user privacy. For example, while the Find Friend and Mobile Tour Guide scenarios could be built on top of these systems, there is no explicit support for managing the privacy issues inherent in these kinds of applications.

[12] In general, positioning systems fall into one of three categories [132]. In the *network-based* approach, infrastructure receivers such as cell towers track cellular handsets or other mobile transmitting units. In the *networked-assisted* approach, location determination occurs in the network with the mobile device's active participation—for example, Qualcomm's Enhanced 911 solution uses handsets to receive raw GPS satellite data that it sends to network processors for calculation. In the *client-based* approach, mobile devices autonomously compute their own position, as is the case with a GPS unit. In this dissertation, we are focused on decentralized systems that make use of the client-based approach because it provides a more solid foundation on top of which stronger guarantees can be made.

A key design decision behind Confab is to place all three of these layers on the end-user's computer rather than distributing them throughout the network infrastructure as in previous approaches (for example, as was done in the PARCTab system [129], the Context Toolkit [45], iRoom [85], and in network-assisted versions of E911 where devices send information to a centralized server for location calculations [132]). In other words, *applications built on top of Confab are structured such that end-users have personal information captured, stored, and processed on their computers as much as possible, and are provided better user interfaces for managing the flow of personal information to others*. This approach gives end-users a greater amount of choice, control, and feedback than previous approaches over what personal information is disclosed to others. This approach also gives end-users a simple conceptual model to understand: all of your information is on your device, and you choose when to disclose it to others.[13] For example, with the Find Friend scenario, an end-user chooses whether to share their location information with the server. With the Mobile Tour Guide scenario, an end-user can choose what level of disclosure she wants to share (ex. city, neighborhood, or street level).

An important issue to address here is the feasibility of this kind of architecture, specifically whether we can expect a great deal of privacy protection at the physical /

[13] Our user studies, described in section 6.2, provide preliminary evidence that this is the default conceptual model that people have for non-badge location-aware systems.

sensor layer. We believe that there will be a useful and non-trivial subset of ubicomp applications built along these lines, for two reasons. First, over the past few years, the research community has been moving from centralized location-tracking architectures (e.g., Active Badge [144]) to decentralized location-support ones (e.g., Cricket [122], Active Bats [146], and Place Lab [131]) for reasons of robustness, scalability, and privacy. We believe that future research will continue this trend in providing privacy protection in the physical / sensor layer for other forms of personal contextual information. Second, there is already a large market for personal items in which sensors can be cheaply embedded, such as PDAs, home security systems, and cars. Although Confab could be used in cases where data is initially captured by others (e.g., by smart rooms or surveillance cameras), we do not explicitly address those cases. Fewer guarantees about the flow of personal information can be made if the data starts outside of one's control. A discussion of the tradeoffs in Confab's architecture is presented in Section 5.8.

Dataflow for Managing Privacy

The dataflow in Confab can be broken down across three major parts: sources, infospaces, and apps (see Figure 5-1). The *infospace* is part of the infrastructure layer and contains contextual information about a person, such as their name, their current location, and their current activity.

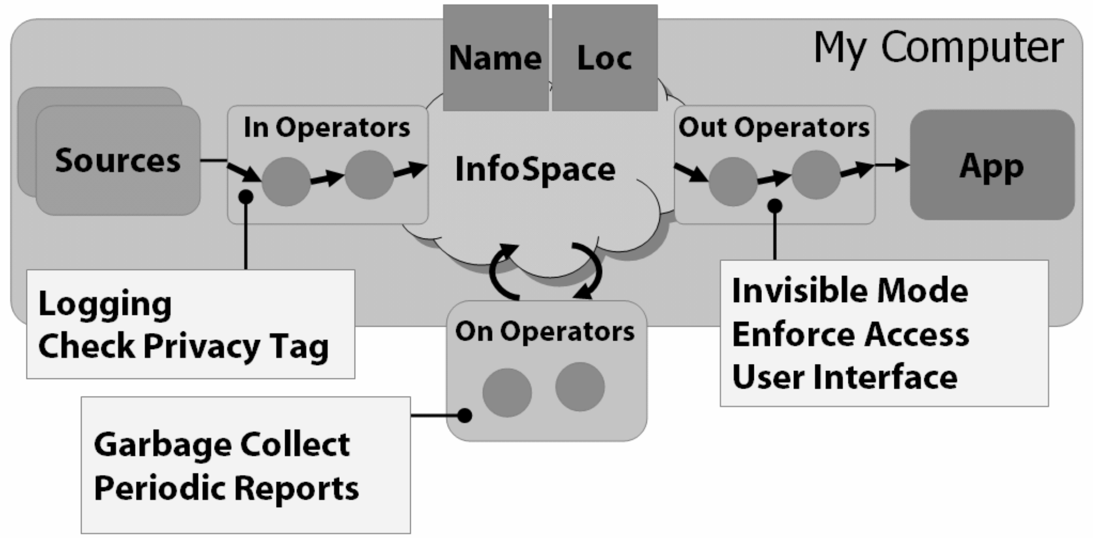


Figure 5-1. The general dataflow in Confab. *Sources* add data to an infospace, a personal repository of information about that individual. As data flows in, it goes through a series of in-operators to manage that flow. Applications can request data from an infospace. This data goes through a series of out-operators before flowing out. On-operators are run periodically on an infospace, for example to generate reports and to do garbage collection of old data.

Sources are part of the physical / sensor layer and are responsible for streaming data into an infospace, for example updating one's location or activity information. Incoming data goes through a series of operators, pieces of composable and reusable code, that manage the flow of data. Operators are a useful abstraction as they allow developers to add or remove functionality without having to modify the main body of source code. Since these operators manage incoming data, they are called in-operators. Some example in-operators currently built into Confab include logging and checking the privacy tag. Privacy tags, discussed in more detail in the next section, are a simple form of digital rights management on data that can specify things like "delete me after five days". For example, in the Find Friend scenario, an end-user could specify that the server should only retain her location information for 72 hours.

Applications can also request data from an individual’s infospace. As the data flows out of an infospace, it goes through a series of out-operators which manage the flow of outgoing data. Some example out-operators include invisible mode (i.e. no information goes out to anyone), enforcing existing access policies, and calling up just-in-time user interfaces. Applications lie outside of the layered architecture described in the previous section. In other words, applications make use of Confab but are not part of it. For reasons of security and privacy, access to an infospace is currently restricted to applications running on the same machine (i.e., localhost).

An infospace also has on-operators that run periodically, including garbage collection of old data and periodic reports. Infospaces, sources, and operators are all run locally, so that no information is disclosed unless the end-user so chooses.

5.3 Confab’s Data Model

Confab’s data model is used to represent contextual information, such as one’s name, location, and activity. People, places, things, and services (entities) are assigned *infospaces*, network-addressable logical storage units that store context data about those entities (see Figure 5-2). For example, a person’s infospace might have static information, such as their name and email address, as well as dynamic information, such as their location and activity.

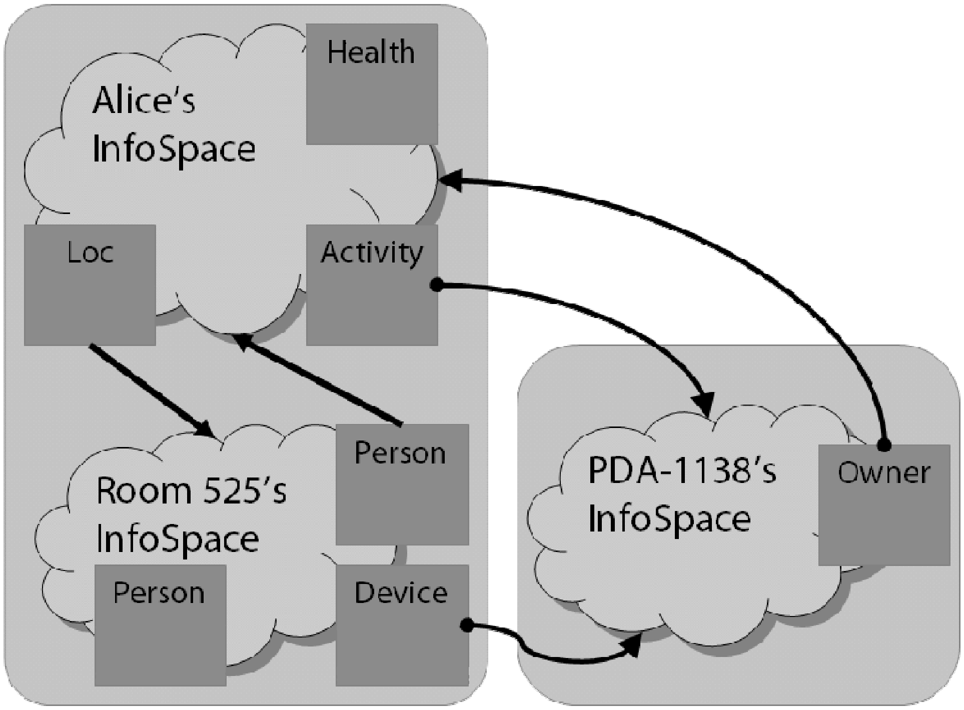


Figure 5-2. An infospace (represented by clouds) contains contextual data about a person, place, or thing. Infospaces contain tuples (squares) that describe individual pieces of contextual data, for example Alice's location or PDA-1138's owner. Infospaces are contained by infospace servers (rounded rectangles). This is the general model for infospace servers and infospaces. For privacy reasons, in this dissertation, we are interested only in the special case where a user's infospace resides on a device owned and managed by that user.

Sources of context data, such as sensors, can populate infospaces to make their data available for use and retrieval. Applications retrieve and manipulate infospace data to accomplish context-aware tasks. Infospaces also provide an abstraction with which to model and control access to context data about an entity. For example, individuals can specify privacy preferences for how their infospace handles access control and flow (described in greater detail in the next section).

Infospaces are managed by *infospace servers*, which can be either distributed across a network or managed centrally, analogous to how a person could choose to have their personal web site hosted on their home machine or by an ISP. Here, for reasons of privacy, we focus on the case where infospaces represent contextual information about individuals and are hosted on devices owned by those individuals.

The basic unit of storage in an infospace is the *context tuple*. Tuples are used to represent intrinsic context, that is an attribute about an entity (e.g., a person's age), as well as extrinsic context, which is a relationship between two entities (e.g., a person is in a room). Tuples are also used to represent static pieces of contextual information (e.g., an email address), as well as dynamic contextual information (e.g., a person's location). These different kinds of contextual information are summarized in Table 5-2.

	Intrinsic	**Extrinsic**
Static	Name, age, email address	A room is part of a building
Dynamic	Activity, temperature	A person is in a specific room

Table 5-2. Confab supports different kinds of context data. Static context data does not change or changes very slowly, whereas dynamic context data changes often. Intrinsic context data represents information about that entity itself, whereas extrinsic context data represents information about an entity in relationship to another entity.

```
<ContextTuple dataformat="edu.school.building"
                datatype="location"
             description="location of an entity"
             entity-link="http://myhost.com/~jdoe"
             entity-name="John Doe"
       timestamp-created="2003.Feb.13 16:06 PST">

   <Values>
      <Value value="523" />
   </Values>

   <Sources>
      <Source datatype="location"
                  link="http://localhost/map.jsp"
                source="Location Simulator"
             timestamp="2003.Feb.13 16:06 PST"
                 value="523" />
   </Sources>

   <PrivacyTags>
      <Notify value="mailto:addr@mail.net" />
      <TimeToLive value="1 day" />
      <MaxNumSightings value="5" />
      <GarbageCollect>
         <Where requestor-location="not edu.school.building" />
      </GarbageCollect>
   </PrivacyTags>

</ContextTuple>
```

Figure 5-3. An example context tuple. Tuples contain metadata describing the tuple (e.g., dataformat and datatype), one or more values, one or more sources describing the history of the data and how it was transformed, and an optional privacy tag that describes an end-user's privacy preferences. In this example, the privacy tag specifies a notification address, a maximum time to live, the maximum number of past values that should be retained, and an additional request to delete the data if the requestor is not in the specified location.

Attributes of interest common to all tuples are *datatype*, a textual name describing the relationship of a tuple to the containing infospace's entity (for example, location or activity); *dataformat*, a string that describes the meaning of the data (for example, temperature could be Farenheit or Celsius); an optional *entity-link* denoting the address of an infospace for an entity described by the tuple; and one or more *values*, each identified by name (see Figure 5-3 for an example). Infospaces can store tuples containing arbitrary data, many of which may describe other entities related to the original infospace. Such tuples' entity-link attributes refer to the infospace of the other entity. For instance, the infospace for a specific room may contain numerous tuples of type 'occupant', each with values denoting a name and email address of an occupant of the room and an entity-link referring to the infospace that hold tuples on behalf of that occupant.

Each tuple can also have an optional attribute called a *privacy tag* that describes hints provided by the end-user on how that tuple should be used when it flows to a computer outside of the end-user's direct control. The current implementation of privacy tags provides hints on when a tuple should be deleted, to help enforce limited data retention. End-users can have their tuples tagged with a *TimeToLive*, which specifies how long data should be retained before being deleted; *MaxNumSightings*, which specifies the maximum number of previous values that should be retained (for example, a value of 5 means only retain the last five places I was at); *Notify*, which specifies an address to send notifications of second use to; and *GarbageCollect*, which specifies additional hints on when the data should be deleted, for example, when the current holder of the tuple has left the area.

By default, when a tuple of any datatype is requested, its value is "UNKNOWN," regardless of whether it actually exists or not. Requests can see correct tuple values only if they have been explicitly granted access. This approach provides some level of plausible deniability, as a datatype might be unknown due to technical failures, lack of actual data, restricted access, or because the person is invisible[14]. It should be noted that while Confab uses a rule-based approach as one underlying mechanism for managing access to one's personal information, the user interfaces are set up such that users do not have to configure preferences a priori. Instead, Confab uses a combination of pessimistic, mixed, and optimistic modes for sharing, avoiding many of the problems with rule-based systems noted in Chapter 2 and Chapter 4.

To help provide a clearer mental model for programmers and end-users, Confab's data model is strongly related to the web (see Table 5-3). Infospace servers are analogous to web servers, infospaces to web sites, and context tuples to web pages. Like a web server, an infospace server represents a unit of administration and a unit of deployment. Like web sites, an infospace represents a unit of ownership and a unit of addressing. Like a web page, a context tuple represents a unit of storage and a unit of data that can be transferred, and can also point to other infospaces.

Furthermore, all of Confab's data model is implemented on top of existing web technologies. Infospace servers are currently implemented on top of the Apache Tomcat web server. Individual infospaces are addressed via URLS, and can be thought of as web-based XML tuplespaces with specialized constraints. Context tuples are

[14] One interesting drawback of making tuples "UNKNOWN" by default is that it makes debugging harder. If a given query does not return the expected results, the built-in ambiguity makes it harder to understand why.

currently represented as data-centric XML documents. That is, context tuples consist only of XML tags and XML attributes, with no text between tags. A thinner version of Confab's infospace has been implemented in C++, for use on smaller devices such as cell phones and PDAS.

Confab	Role	Web analogy
InfoSpace Server	Manages a collection of InfoSpaces Unit of administration Unit of deployment	Web server
InfoSpace	Manages a collection of Context Tuples Represents a single entity Represents a zone of protection Unit of ownership Unit of addressing	Web site
Context Tuple	Represents information about an entity Contains privacy preferences Unit of storage	Web page

Table 5-3. This table summarizes the three main concepts of Confab's data model.

In summary, Confab's data model can be broken up into three primary components: infospace servers, infospaces, and context tuples. Infospace servers manage a collection of infospaces, and infospaces manage a collection of context tuples. Context tuples represent an individual piece of information about an entity (a person, place, thing, or service). Context tuples also optionally contain a privacy tag that contains rules on how that tuple should be used and when it should be deleted.

5.4 Confab's Programming Model

There are six major components that developers can directly use in developing privacy-sensitive ubicomp applications, namely operators, Confab Client, active properties, service descriptions, access notifications, and the Place Bar. A brief description of each is provided in Table 5-4, with a more detailed description below.

Component Name	Description
Operators	Manages the flow of data going in or out of an infospace
Confab Client	Simple client that makes requests to an infospace
Active Properties	Wrapper around Confab Client for maintaining fresh data
Service Descriptions	Describes what information an applications wants and what options are provided to end-users. Can be coupled with Access Notifications and the Place Bar to provide just-in-time decisions.
Access Notifications	Presentation layer support for pull transactions, ones where others request information from you first
Place Bar	Presentation layer support for push transactions, ones where end-users choose to send information first

Table 5-4. An overview of the components provided in Confab for application developers.

5.4.1 Infospace Operators

Many of the design decisions about Confab were drawn from the architecture of the World Wide Web, or as Fielding calls it, the REST (Representational State Transfer) architectural style [54] for large-scale distributed hypermedia systems. One reason HTTP has succeeded is because there is a relatively small but useful set of verbs (for example, GET and POST) that can be applied to a large number of nouns (for example, HTML, GIF, JPG, PDF, and so on) in a stateless manner. This keeps implementation relatively simple, minimizes the state that must be kept between requests (thus making the system easier to scale), requires minimal processing power by servers and clients (as demonstrated by the number of tiny servers and thin clients available), and perhaps most importantly, is backwards as well as forwards compatible. That is, the verbs in the HTTP protocol are simple to implement and cover a wide-enough range of functionality such that it is unlikely that new verbs will be needed. Future clients will still be able to work with existing servers, and existing clients will still be able to work with future servers, or more succinctly, things will still work properly even after upgrades.

The simplicity of this approach is in sharp contrast to Jini, CORBA, and other alternatives in distributed computing which aim for distributed objects (data coupled with code) rather than distributed multimedia content (data). While in theory distributed objects can provide richer semantics, it also requires intimate knowledge of the API of those objects, making it hard to maintain compatibility between versions and thus difficult to achieve the positive network effects and economies of scale that occur when large numbers of people use the same system. These alternatives also usually require a non-trivial amount of processing power and storage, making them harder to run on thin clients.

Confab's infospaces follows the same philosophy as HTTP. Infospaces support two general kinds of methods, *in* and *out*. In-methods affect what data is stored within an infospace, and include ADD and REMOVE. As suggested by the similar naming, in-operators are activated only for in-methods. Out-methods govern any data leaving an infospace, and include QUERY, SUBSCRIBE, UNSUBSCRIBE, and NOTIFY. As above, out-operators are activated only for out-methods. All of these methods combine to form the small set of verbs that can be applied across a variety of nouns, namely context tuples that represent a range of personal data about individuals[15]. As with the Web, this approach has the advantage of having a relatively simple implementation, and in theory should improve compatibility. These methods are currently implemented as extensions of the HTTP protocol. For example, a client can connect to an infospace and then make a request to add a tuple or query for tuples matching a given datatype and dataformat.

Each infospace also contains *operators* for manipulating tuples. Operators are chainable pieces of code that can be added to an existing infospace to extend and customize it to what is needed without having to modify the main body of code. Confab supports three different kinds of operators: in, out, and on. *In-operators* are run on all tuples coming in through in-methods. An example in-operator is one that checks the infospace's access control policies to make sure that this is a tuple that is allowed to be added. *Out-operators* are run on all tuples going out through out-

[15] Currently, only ADD, REMOVE, and QUERY are fully implemented. Partial implementations of SUBSCRIBE, UNSUBSCRIBE, and NOTIFY are available. However, we are debating whether these should be part of the infospace protocol, because they increase the state that an infospace must maintain (raising implementation complexity and making it more difficult to have smooth restarts in case of failure) for relatively little benefit.

methods. An example out-operator is one that blocks all outgoing tuples if the user is in invisible mode. *On-operators* are operators that run periodically, such as garbage collection. Table 5-5 shows a full list of operators provided in Confab by default.

Operator Type	Description
In	Enforce access policies Enforce privacy tags Notify on incoming data
Out	Enforce access policies Enforce privacy tags Notify on outgoing data Invisible mode Add privacy tag Interactive
On	Garbage collector Periodic report Coalesce

Table 5-5. Confab provides several built-in operators, which can be added or removed to modify what a tuple contains and how it flows to others. In-operators manage the flow of incoming data, while out-operators manage outgoing data. On-operators are run periodically.

The two Enforce Access Policies operators (in- and out-) let end-users specify access policies for their infospace. Several different conditions can be specified for authorization, including who is requesting the data, what data they are requesting, how old the data is, what Internet domain or IP address they are requesting from, as well as the current date and time.

The two Enforce Privacy Tags operators are used to put the preferences specified in privacy tags into action. The out-operator version makes sure that data that should not leave an infospace does not, while the in-operator version does the same with incoming data. Together, a set of infospaces can provide peer enforcement of privacy tags, helping to ensure that data is managed properly (see Figure 5-4). If tuples are digitally signed, peers can also check if privacy tags have been altered, thus detecting that an infospace is not handling personal information properly. However, this feature is not yet implemented in the current version of Confab.

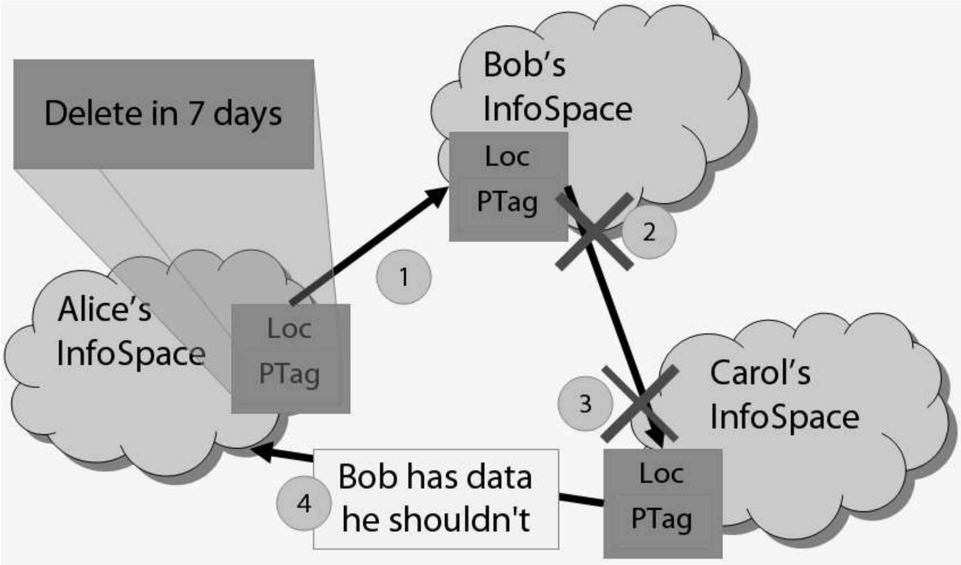


Figure 5-4. An example of peer enforcement. (1) Alice shares her location with Bob, which is tagged to be deleted in 7 days. Suppose 7 days have passed, and that Bob passes the data on to Carol. If this is an accidental disclosure, then (2) his infospace prevents this from occurring. If intentional, then (3) Carol can detect that Bob has passed on data that he should not have, and (4) notifies Alice.

The Notify operators are used to send short messages to give end-users feedback about who is requesting information and when. Notify operators can currently be configured to send messages either through email or via instant messenger.

The Invisible mode operator can be used to block all outgoing tuples and return the value of "UNKNOWN" to all queries. It can also be configured to return some pre-specified value, allowing users to make “white lies.” The Add Privacy Tag operator is used to add end-user or application defined privacy tags to outgoing tuples.

The Interactive operator can be used to give end-users control over disclosures. In the current implementation, when a request comes in and the Interactive operator is active, a simple GUI is displayed, giving the end-user several options, including disclosing the requested information just this once, ignoring it, or denying access permanently. An example of this user interface is shown in Figure 5-7.

The Garbage Collector operator is run periodically to delete any context tuple that has a privacy tag specifying that it should be deleted. The Periodic Report operator sends an email to the owner of an infospace, providing a periodic summary of who has requested what (e.g., every day, week, or month). The Coalesce operator is used to delete tuples with repeated values, providing a more compact representation of one’s history. For example, suppose a user has a sensor that updates her infospace with her current location information every minute. If she has not moved for an hour, there will be sixty tuples with the exact same location value. Here, the Coalesce operator simply sorts all of the location tuples by time and deletes tuples with duplicate values, keeping only those needed to determine when she entered and exited a location.

Operators are loaded through a configuration file on startup, and are executed according to the order in which they were added. Each operator also has a filter that checks whether or not it should be run on a specific tuple. When an in- or out-method is called, a chain of the appropriate operators is assembled and then run on the set of incoming or outgoing tuples.

Note that peer enforcement and automatic deletion of old data can be trusted to execute on computers that an end-user has control over, but not necessarily on computers owned by others. Short of a widely deployed trusted computing base (which itself poses serious privacy risks), there is no way of forcing others to delete data. Privacy tags let end-users provide a hint saying what their privacy preferences are, and relies on social, legal, and market mechanisms that others will do the right thing. In cases where there is a strong level of trust, this will suffice and can help prevent accidental disclosures. In cases where there is not a great deal of trust, other mechanisms, such as passing on coarser-grained data or anonymity, should be used.

5.4.2 Confab Client and Active Properties

Confab also comes with a simple Confab Client implemented in Java aimed at facilitating an application developer's interactions with an infospace. This client provides a simple layer of abstraction for adding, removing, and querying tuples, so that programmers do not have to be concerned with the vagaries of Confab's network protocol for interacting with infospaces.

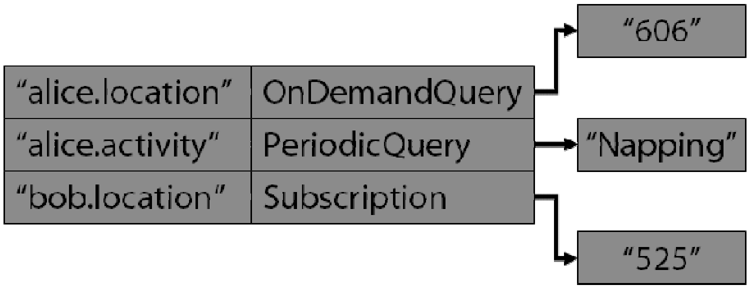


Figure 5-5. Clients can maintain a list of properties they are interested in through an Active Properties object, which will automatically issue queries and maintain last known values.

To simplify the task of querying for and maintaining context state in applications, Confab provides an *active properties* class (see Figure 5-5). Queries can be placed in an active properties instance and be automatically executed to get up-to-date values. These queries can be given semantically useful names by the programmer, for example `alice.location` or `alice.activity` . Last known values are also cached to provide a level of fault-tolerance should the requestor or requestee be temporarily disconnected. This also reduces the load on an infospace server.

Active properties supports three different kinds of properties: OnDemandQuery, which makes a request for new data whenever its value is checked; PeriodicQuery, which periodically checks for new data; and Subscription, which periodically receives new data from an infospace. After an active properties instance has been initially set up, an application can get a value simply by using the property name (e.g., `alice.location`) to retrieve the last-known value.

5.4.3 Service Descriptions

Applications can publish service descriptions that describe the application, what information the application needs, as well as various options that end-users can choose from. For example, Scenario 2 at the beginning of this chapter described a mobile tour guide service that offered different kinds of information depending on the precision of information Alice was willing to share.

Confab provides support for applications to specify these different options, as shown in Figure 5-6. These service descriptions provide basic information about the service, for example the name of the service and a URL for more information. Service descriptions can also contain options that describe what features that option offers, what datatypes and dataformats are needed from the end-user, and how often the information will be queried.

When an application first makes a request to an infospace, it sends its service description. If the infospace has seen this service description before, it simply uses a previously stored configuration associated with that description, which specifies whether to allow access and what option to use[16]. Currently, service descriptions are uniquely identified by using the service provider, service name, and version number to generate a key.

[16] As Confab is currently implemented on top of web technologies, identification is done through cookies. Note that this is different from how cookies are normally used, in that these cookies are used exclusively on the local machine to identify applications, rather than being transferred across the network to identify users.

```
<Service    name="Tourguide"
     description="Tourguide for cities"
       keywords="Tourism, Location"
       provider="Bob Inc"
            url="http://bob.com/tourguide"
        version="1.0">

 <Option   name="1"
     dataformat="city"
       datatype="location"
         method="get"
          offer="Events, Museum lines"
           rate="15 minutes"
       timespan="current" />

 <Option   name="2"
     dataformat="zipcode"
       datatype="location"
         method="get"
          offer="Stores, Recommendations"
           rate="30 seconds"
       timespan="current" />

 <Option   name="3"
     dataformat="latlon"
       datatype="location"
         method="get"
          offer="Route Finder, Real-time map"
           rate="30 seconds"
       timespan="current" />

</Service>
```

Figure 5-6. Confab's service descriptions allow services to give end-users various choices when using a service. This example shows the service description for a mobile tour guide service. The first option (where name="1") provides information about events and the length of museum lines in the city. To do this, the service needs the end-user's current location at the city level every 15 minutes.

If the infospace has not seen this service description before or the previous settings have expired (for example, if the user only gave the application access for one day), a default access notification GUI is displayed which lets end-users choose whether to allow access, what option they want, and how long the settings should last (described below, see Figure 5-7). This approach gives service providers a way of giving end-users flexibility over what features they are interested in using as well as what privacy tradeoffs they are willing to make.

5.4.4 Access Notification User Interface for Pull Transactions

Confab provides GUI notifications to let end-users make just-in-time decisions for pull transactions, which are transactions initiated by others, for example when a friend requests your current location. The challenge in creating a user interface for managing end-user privacy is in providing end-users with a simple, understandable, and appropriate level of visibility and control. Potential design flaws here include having a muddled conceptual model of what information is flowing where, providing too little information so as to make it difficult to make good decisions, and overwhelming end-users with too many options or too many notifications.

The current notification user interface was developed over four iterations with seven people. This design of this user interface was also informed by the results of the surveys and interviews described in Section 2.2.4, as well as the pitfalls in user interfaces for privacy described in Chapter 4. Here, we use the term "survey" to refer to the freeform comments in the survey that led to the design of the Faces user interface, and "interview" to refer to the interviews conducted with twenty people to understand their interests and concerns about location-enhanced services.

There are three design points that can be drawn from the surveys and interviews. First, people distinguished between sharing personal information with other people versus sharing with services. Second, temporal boundaries were often used to circumscribe whether or not personal information was shared with others. Third, people distinguished between sharing information continuously versus sharing it discretely. We describe each of these in more detail below.

People Distinguished Between Sharing with People and with Services

Unsurprisingly, many people distinguished between sharing information with other *people*, such as a co-worker or a spouse, versus sharing information with *services*, such as a map service or emergency response service. For example, one person from the survey wrote: “I would never want a retailer to contact me unasked, but always want my spouse to find me.” Another wrote: “My significant other should see my trueface always. The evil national chains should see my blank face always.”[17]

What is interesting here is how people described their sharing preferences. In both cases, people are making risk, benefit, and trust judgments about how their personal information will be used if it is shared. The key observation here, however, is that when sharing with other *people* these dimensions seem to be implicitly collapsed simply into who that individual is and what their role or relationship is,

[17] The terms “trueface” and “blank face” refer to the metaphor used in the survey, as described in section 2.2.4. True face means all information about an individual, blank face means no information.

whereas with *services* users make a judgment of the perceived risk and benefit of that service, requiring that these dimensions be spelled out more explicitly. The design implication here is that a UI for sharing information with a service should provide more information than one for sharing information with another person, and include enough details about the service so that people can make good decisions.

For example, with respect to sharing with other people, one person from the survey wrote: “I can use my degree of trust in people to determine what they should know about me.” Another wrote: “I don’t mind if my friends and family know where I am and what I am up to.” A third wrote: “If I don’t trust the person with personal information, I wouldn’t want to give them any information at any time. If I do trust the person, I’m willing to give out information freely.”

In contrast, with respect to services, interviewee #5 said about location-based advertising and shopping support: “Sometimes advertisements are helpful. Sometimes you’re watching TV and you’re like oh…1000 anytime minutes. And you’re pushing through junk mail and if you had the time look through. But the more specific and the less you have to disregard, it’s useful. If it’s tailored and you don’t have to fish, it’s all going to be helpful.” Interviewee #6 said: “I don’t know if I want people to give me more advertisements. I think it’s okay if they know that they know I like Mexican food. And it would be useful if say the Mexican restaurants pop up at the top.”

With respect to the emergency response application, interviewee #7 said: “This is really a privacy issue then. They’ll give you an ID or something. They say they won’t use it for any other purposes. But it’s just a Big Brother. If it was secure, then it’s useful. Useful to locate where everyone is located within a building.” Interviewee #15 had a similar response, saying: “What I am afraid is the government will know too much. It may be abused or misused by the government. When you say emergency, how often do you encounter that? It seems to be very useful at an emergency. Emergencies don’t happen daily. It may not be a good idea. I personally wouldn’t want them to know.” However, interviewee #14 had a different perspective on the utility of the emergency response application, saying: “This is life and death, then I should. So absolutely, I would be very anxious to disclose information. If I am at work, I would give the exact location and how to get to me. The routes. Exact information as possible. When you deal with life or death, then it does not cost any more extra time to release any information.”

Again, the main point here is that with respect to services, people judge whether they want to share information based on a variety of factors, in contrast to whether or not information is shared with other people. Thus, it is important for services to provide enough information to help people make good choices.

People Used Temporal Boundaries to Manage Access to Personal Information

In the surveys and interviews, when describing their preferences for sharing information in natural language, people often used temporal boundaries to help manage access to personal information. One way this was done was to limit when access was allowed. For example, a typical comment from the survey was, “during the work day, or after-hours during crunch time, I’d want my boss/coworkers to find m[e] - after hours I’d rather be more anonymous.” Another survey responder wrote, “Wouldn’t want to share with coworkers/bosses. If it’s during a work day and we are trying to get something done, then it’s useful. When I leave for the day, that the device is off.” In many cases, these preferences can be described as filling in the blanks of who can access information and when. For example, using a response from one of the survey responders, “Work people can know my information during work hours. Home/SO people can know my information always.”[18]

Another way time can be used is to allow temporary access. For example, interviewee #1 observed that temporary access would be useful, “if friends are in

[18] Interestingly, no person from the survey or the interviews used granularity as an option when describing in plain language their preferences for sharing personal information with other individuals. Informal discussions with other researchers suggest that while granularity can be useful a technique when sharing personal information with services, it is probably not effective when sharing with people because there is a lack of plausible deniability for the granularity to be so coarse, would probably signal that the person is not privileged (removing another level of plausible deniability), and would also likely annoy the person asking. In other words, the majority of people seemed to prefer an all or nothing scheme.

One interesting research direction suggested by a participant during the user studies was that the granularity of location information returned could be based on the distance between the two people, as that would be more likely to be semantically useful. For example, it might be more useful to return city level information if the person asking were 5000 kilometers away, but room level if the person asking was in the same building. The granularity returned could also be based on the relationship between the two parties. For example, a co-worker might only care if someone is in the office or not, while a father might care if someone is at school or home.

town, if there is a researcher I want to know, if there is a conference in town." Temporary access is useful in cases where there are limited interactions between people, for example with a new acquaintance or with a friend that is visiting, letting people share personal information without having to remember to remove access privileges later on.

The design implication here is that user interfaces should support the use of time to help people manage their personal privacy, both in terms of limiting and allowing access, as well as providing temporary access.

People Distinguished Between Continuous and Discrete Access to Information

As noted earlier in Section 2.3, people had different reactions to continuous disclosures of information (for example, a parent continuously tracking a child) versus discrete disclosures (for example, a co-worker checking if a colleague is in). Concerns were raised primarily about continuous disclosures. For example, with respect to the active campus applications, interviewee #3 commented, "It's stalking, man." Interviewee #2 echoed the same concerns, saying, "I would be creeped if my friends found me. And they said *I saw you here*. It would just be weird." Interviewee #9 had similar concerns, saying "I wouldn't mind, but I don't know how useful it would be. I don't know if everyone wants to know where I am every second."

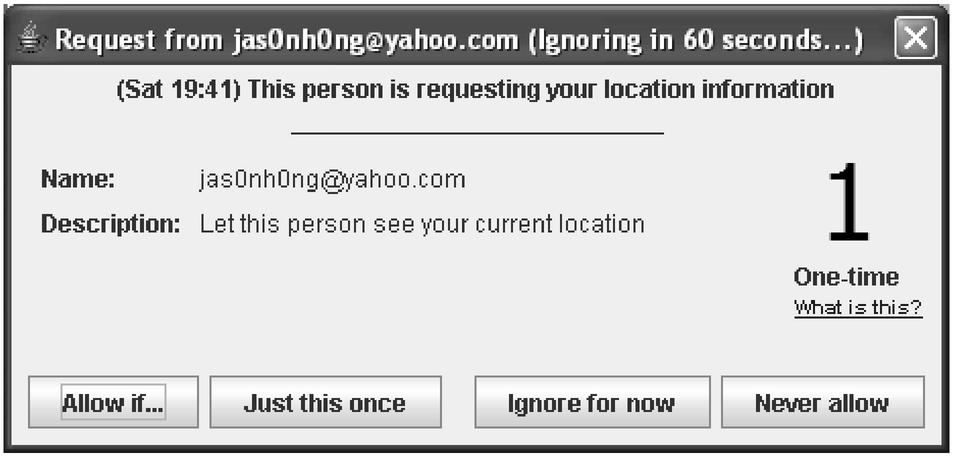
This does not mean, however, that there are no good reasons for continuous disclosures. One could imagine a health monitoring system that continuously shares information with a doctor, or parents that continuously share location information with one another for the next hour so that they can coordinate better. The primary design implication here is that user interfaces should make it easy to differentiate between discrete versus continuous disclosures of information.

Interacting with Access Notifications

In summary, responses from the surveys and interviews suggest three different design guidelines:

- User interfaces for sharing personal information with services need to have more details about how the information will be used than user interfaces for sharing personal information with other people.
- User interfaces should support the use of temporal boundaries in managing personal privacy.
- User interfaces should make it easy to differentiate between personal information that is shared continuously versus discretely.

Figure 5-7 shows the current implementation of the access notification user interface. More specifically, it shows a request from a person. The center of the notification shows who is making the request ("jas0nh0ng@yahoo.com") as well as a short description of why the request is being made (in this example, the notification simply shows the default description, “Let this person see your current location”, but it could also show a text message from the requestor).



Request from jas0nh0ng@yahoo.com (Ignoring in 60 seconds...)

(Sat 19:41) This person is requesting your location information

Name: jas0nh0ng@yahoo.com

Description: Let this person see your current location

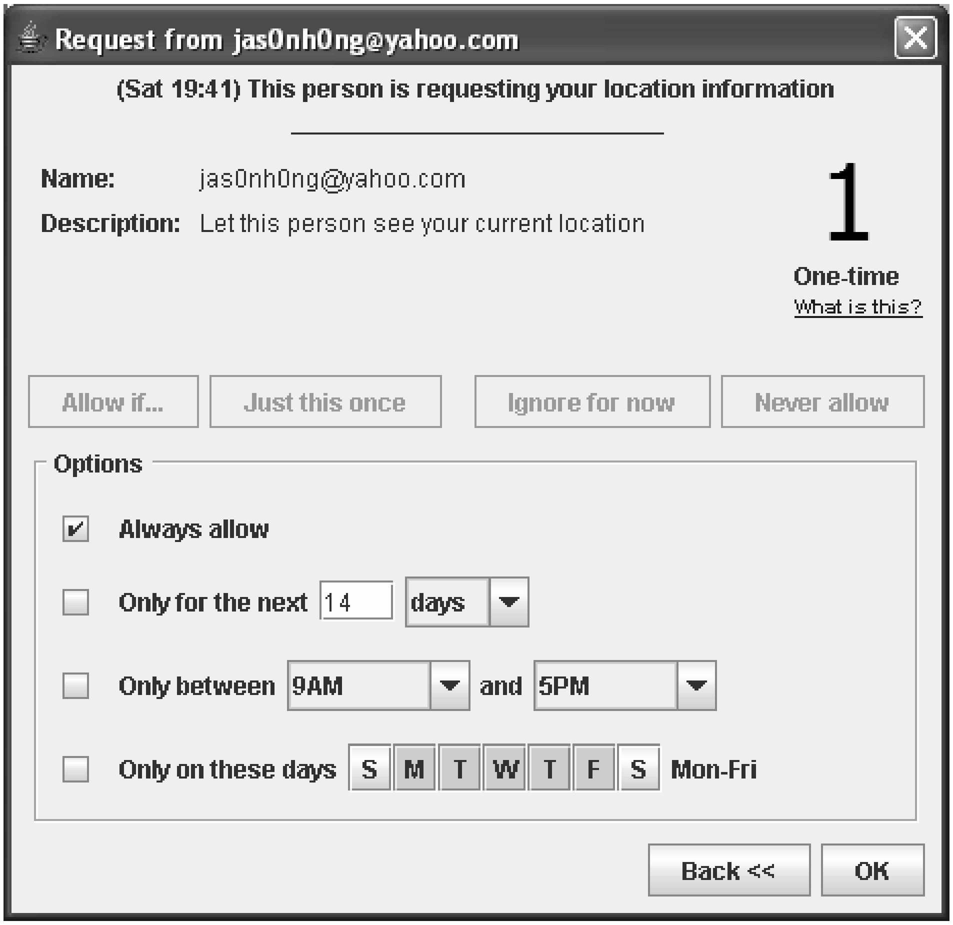
Figure 5-7. Access notifications are just-in-time descriptions of who is requesting information and why. The large “1” on the right signifies that this is a one-time disclosure, as opposed to a continuous disclosure of information. The buttons along the bottom let people choose to share information or to ignore. The “Allow if…” button shows additional options, as shown in Figure 5-8. If the notification is ignored (which happens if the user hits the ignore button or if the notification times out) then a reply of “UNKNOWN” is returned, helping to ensure some level of plausible deniability.

The top of the notification shows the name of the requestor, and also indicates how much time is left before the notification is automatically ignored. To minimize distraction, the number of seconds is updated every 10 seconds rather than every second. If this notification is ignored, it returns a value of “UNKNOWN” to the requestor.

The large number “1” on the right side indicates that this notification is a one-time disclosure. A large infinity symbol “∞” would indicate that this is a repeated and continuous disclosure. Hovering over or clicking on the “What is this?” on the right side would provide more information to help people understand the difference between these two.

There are also several buttons across the bottom that let people choose whether they want to disclose information or not. The button “Just this once” discloses information just once. The button “Ignore for now” ignores the current request, returning a value of “UNKNOWN.” Note that the system might not actually know the user’s current location, helping to foster a level of plausible deniability. The button “Never allow” means that the requestor will always be ignored and thus always see a value of “UNKNOWN.” The button “Allow if…” brings up additional options that are less frequently used (see Figure 5-8).

There are several options in this user interface. A user can choose to always allow the requestor to see information (a common preference mentioned for relationships where there is a strong level of trust, such as family and close friends). A user can also choose to give a requestor temporary access for the next few days or next few hours. Lastly, a user can choose to allow access only between certain times or only on certain days. It is important to note here that users will always get a notification when their information is being requested. These options simply specify when information is automatically disclosed.



Request from jas0nh0ng@yahoo.com

(Sat 19:41) This person is requesting your location information

Name: jas0nh0ng@yahoo.com

Description: Let this person see your current location

1

One-time

What is this?

Options

☑ Always allow

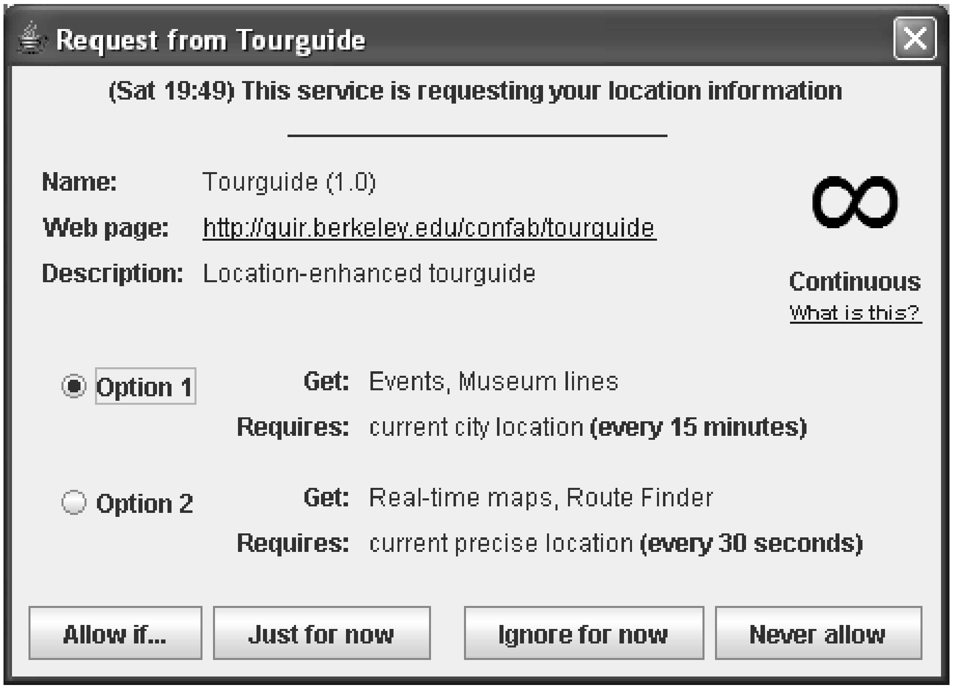
□ Only for the next 14 days

□ Only between 9AM and 5PM

□ Only on these days S M T W T F S Mon-Fri

Figure 5-8. The extended options version of an access notification, which is shown if the user clicks on the "Allow if…" button. Users can choose to "Always allow" access, provide temporary access ("Only for the next 14 days" or "Only for the next 2 hours"), between certain times ("Only between 9AM and 5PM), or only on certain days of the week. This user interface is designed such that either "Always allow" is selected, or any combination of the remaining ones (for example, both "Only for the next 14 days" and "Only between 9AM and 5PM"). Clicking on the Back button returns to Figure 5-7.

Figure 5-9 shows an access notification from a service. This UI lets people choose what features they want based on what level of information disclosure that they are comfortable with. Access notifications for services can be automatically generated from service descriptions, described in Section 5.4.3.



(Sat 19:49) This service is requesting your location information

Name: Tourguide (1.0)

Web page: http://guir.berkeley.edu/confab/tourguide

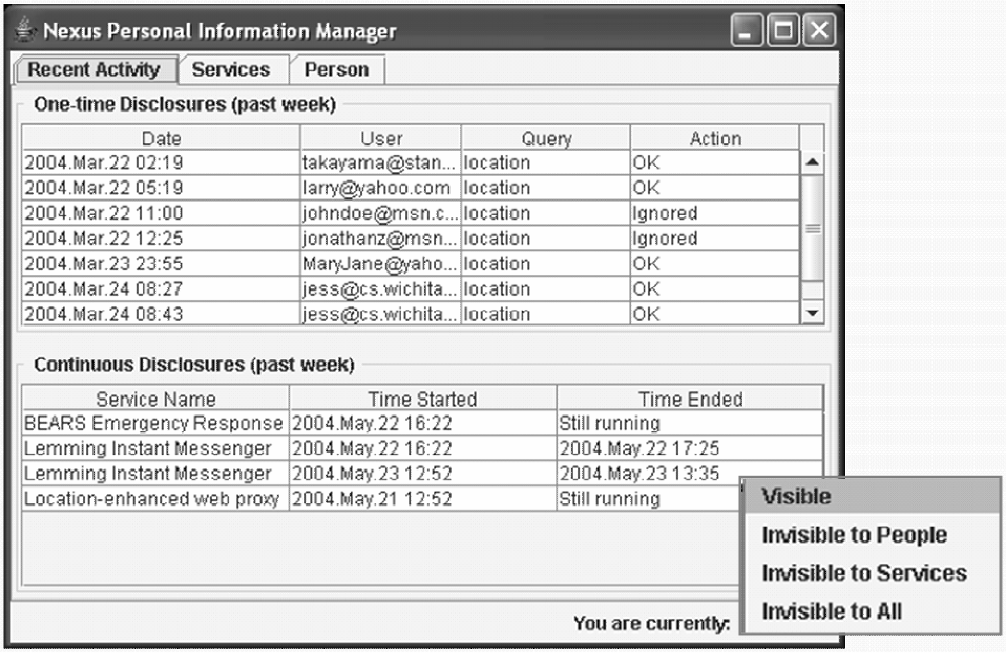
Description: Location-enhanced tourguide

∞ Continuous

● Option 1 Get: Events, Museum lines
Requires: current city location **(every 15 minutes)**

○ Option 2 Get: Real-time maps, Route Finder
Requires: current precise location **(every 30 seconds)**

Figure 5-9. An access notification request from a service. This UI provides details about a service, as well as several options that let people choose what level of information to share and what services they get in return. Here, a user can share her current city location and get events and the length of museum lines, or precise location and get real-time maps and a route finder.



One-time Disclosures (past week)

Date	User	Query	Action
2004.Mar.22 02:19	takayama@stan...	location	OK
2004.Mar.22 05:19	larry@yahoo.com	location	OK
2004.Mar.22 11:00	johndoe@msn.c...	location	Ignored
2004.Mar.22 12:25	jonathanz@msn...	location	Ignored
2004.Mar.23 23:55	MaryJane@yaho...	location	OK
2004.Mar.24 08:27	jess@cs.wichita...	location	OK
2004.Mar.24 08:43	jess@cs.wichita...	location	OK

Continuous Disclosures (past week)

Service Name	Time Started	Time Ended
BEARS Emergency Response	2004.May.22 16:22	Still running
Lemming Instant Messenger	2004.May.22 16:22	2004.May.22 17:25
Lemming Instant Messenger	2004.May.23 12:52	2004.May.23 13:35
Location-enhanced web proxy	2004.May.21 12:52	Still running

You are currently:

Figure 5-10. This UI shows who has requested what information. It also provides a simple way of going into invisible mode for just people, just services, or to everything.

Figure 5-10 shows an example user interface that we have built showing the access notification logs. This user interface is independent of the access notification user interfaces (i.e., Figure 5-7, Figure 5-8, and Figure 5-9), and is meant to help people understand where their personal information is flowing after the fact. It displays one-time disclosures, showing who has requested what information, as well as continuous disclosures, showing what services are active and whether they are still running or not. This user interface also provides a convenient way of going into invisible mode for just people, for just services, or for both people and services.

Avoiding the Pitfalls

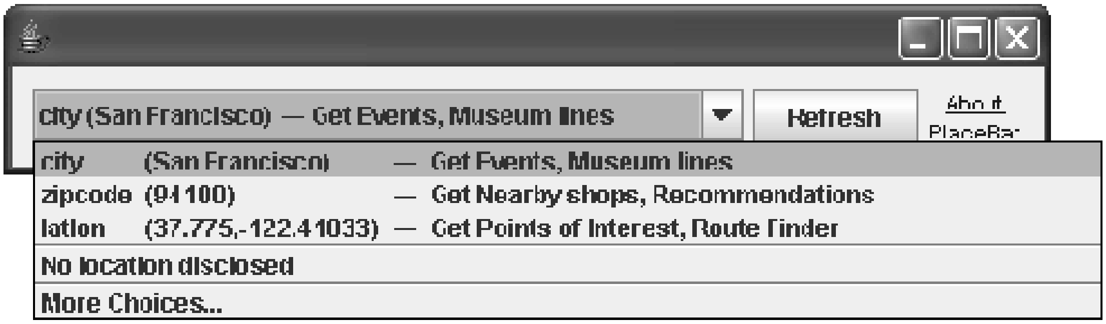
The access notification user interface was designed to avoid the pitfalls in user interfaces described in Chapter 4. It shows the actual flow of information, as a notification is brought up every time someone requests information. The interface is also designed with a minimal amount of configuration, allowing people to make decisions about disclosures as requests happen, rather than having to configure them in advance. It avoids using a strict rule-based system, instead using a combination of pessimistic, mixed, and optimistic modes for simplifying interactions.

With respect to coarse-grained control, the main choices of “Just this once” and “Ignore for now” are shown first, with additional options for limiting access based on time hidden until needed. Lastly, the default information disclosed to requestors is “UNKNOWN.” This could be for technical reasons, or because the person is busy, or because the requestor is in the person’s never allow list. Since requestors cannot discern which of these is the case, this gives people a level of plausible deniability.

User feedback about access notifications is described in the next chapter.

5.4.5 Place Bar User Interface for Push Transactions

Confab also provides a Place Bar user interface widget (see Figure 5-11) to let end-users choose how much information is disclosed for push transactions, which are transactions initiated by the user. The Place Bar was developed after feedback on early iterations of a location-enhanced tourguide suggested that people thought of it as a push application rather than a pull application. The original idea for the Place Bar was co-developed with researchers at Intel [76].



city (San Francisco) — Get Events, Museum lines

Refresh
About PlaceBar
city (San Francisco) — Get Events, Museum lines
zipcode (94100) — Get Nearby shops, Recommendations
latlon (37.775,-122.41033) — Get Points of Interest, Route Finder
No location disclosed
More Choices...

Figure 5-11. The Place Bar is a user interface widget for managing push transactions.

The Place Bar is intended more for sharing information with services than with people. As such, it describes what services are offered at a given level of disclosure and what information is required from people. For example, Figure 5-11 shows a Place Bar configured to the service description of a location-enhanced tourguide. This tourguide lets people get events and museum lines if they share their current city information, nearby shops and recommendations if they share more precise information (in this case, zipcode), and nearby points of interest and a route finder if they share the most precise information possible (in this case, latitude and longitude). The Place Bar also shows the current value of the different options (e.g., "San Francisco" and "94100"), so that users will have a clear idea of what information is being shared.

It should be noted here that while people generally understand the concept of the Place Bar, it proved difficult to use in practice. More details from the user studies are described in the next chapter.

5.4.6 Programming Model Summary

In summary, Confab's data model and programming model provide application developers with a framework and a suite of mechanisms for building privacy-sensitive applications. Operators are used within an end-user's infospace to help control the flow of personal information, and can be customized to fit specific end-user needs. Confab Client and Active Properties provide simple APIS that application developers can use to access and update an infospace. Service descriptions are used by applications to describe what kinds of personal information are needed, as well as at what granularity and at what rate. Access Notifications and the Place Bar provide reusable user interface components for managing the flow of personal information.

5.5 Extensions for Location Privacy

Since location-enhanced applications are a rapidly emerging area of ubiquitous computing, Confab currently comes with specific extensions for capturing and processing location information in a privacy-sensitive manner. In this section, we describe the Place Lab sensor source and the MiniGIS operator for processing location information.

Place Lab Source for Determining Location

Place Lab [131] uses the wide deployment of 802.11b WiFi access points for determining one's location in a privacy-sensitive manner. The key observation here is that many developed areas have wireless hotspot coverage so dense that cells overlap. By keeping a local cache of a Place Lab directory, which maps the unique MAC address of a wireless hotspot to a physical latitude and longitude, mobile computers and PDAs equipped with WiFi can determine their location to within a city block.

Figure 5-12 shows a simple graphical example of how Place Lab works. If a computer knows the geographic location of WiFi access point A and can detect that access point, then it can assume it is within 50-100 meters of that access point, as that is the general range of most WiFi access points[19]. If a computer knows the location of access points B and C and can detect both access points, it can assume it is roughly between the two. It should be noted that Place Lab works even with encrypted access points, because Place Lab relies solely on the ability to detect access point MAC addresses, which are not encrypted, rather than the privilege of using those access points.

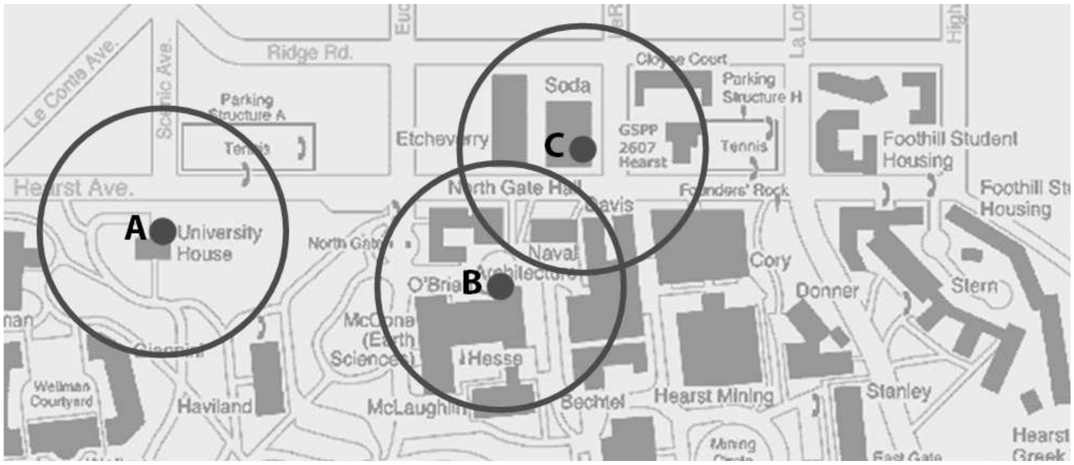


Figure 5-12. Place Lab provides location information in a privacy-sensitive manner at a granularity of roughly 50-100 meters. Devices equipped with the Place Lab database can passively detect the MAC address of known access points and then lookup the location of those access points on a local database. Using the example above, if a device can see access point A, then the device can assume it is within 50-100 meters of that access point. If a device can see both access points B and C, then the device can assume it is in the intersection of the two.

[19] For example, LinkSys' Wireless Technology Comparison Chart [102] lists the range for 802.11b as "Typically 100-150 feet indoors, depending on construction, building materials, room layout."

Place Lab offers many advantages over existing techniques for determining one's location. First, Place Lab works without any special equipment other than a WiFi card. The appeal of this approach becomes apparent as more and more devices come with WiFi integrated into them[20]. Second, Place Lab works indoors and in urban canyons, places where GPS does not always work effectively. Third, since wireless hotspots can be detected passively, computers can determine their location without divulging any information to any third parties or other entities. In other words, Place Lab provides protection at the physical / sensor layer[21].

Lastly, Place Lab takes advantage of two different WiFi adoption trends. First, every month, roughly three to four hundred thousand WiFi access points are sold worldwide [58]. The more access points that are deployed, the higher the precision and the greater the coverage of Place Lab. Second, the location of WiFi access points is already being collected for free by hobbyists known as wardrivers. These hobbyists use WiFi detectors coupled with GPS devices to collect and upload this information to web sites (e.g., wigle.net), which can be freely downloaded[22].

[20] For example, WiFi is integrated into Intel's Centrino chip. Apple iBook laptops and iPaq PDAs are examples of devices with WiFi integrated directly into the device. Some mobile phones have also shipped with WiFi capabilities.

[21] There are interesting tradeoffs here between privacy, consistency and freshness of data, as well as computational and storage requirements of the client. We discuss this as future work in Chapter 8.

[22] Note that the legality of publishing the geographic location of WiFi access points is not clear at this point. For example, California Penal Code 502(c)(6) states:

> "Any person who commits any of the following acts is guilty of a public offense: ... Knowingly and without permission provides or assists in providing a means of accessing a computer, computer system or a computer network"

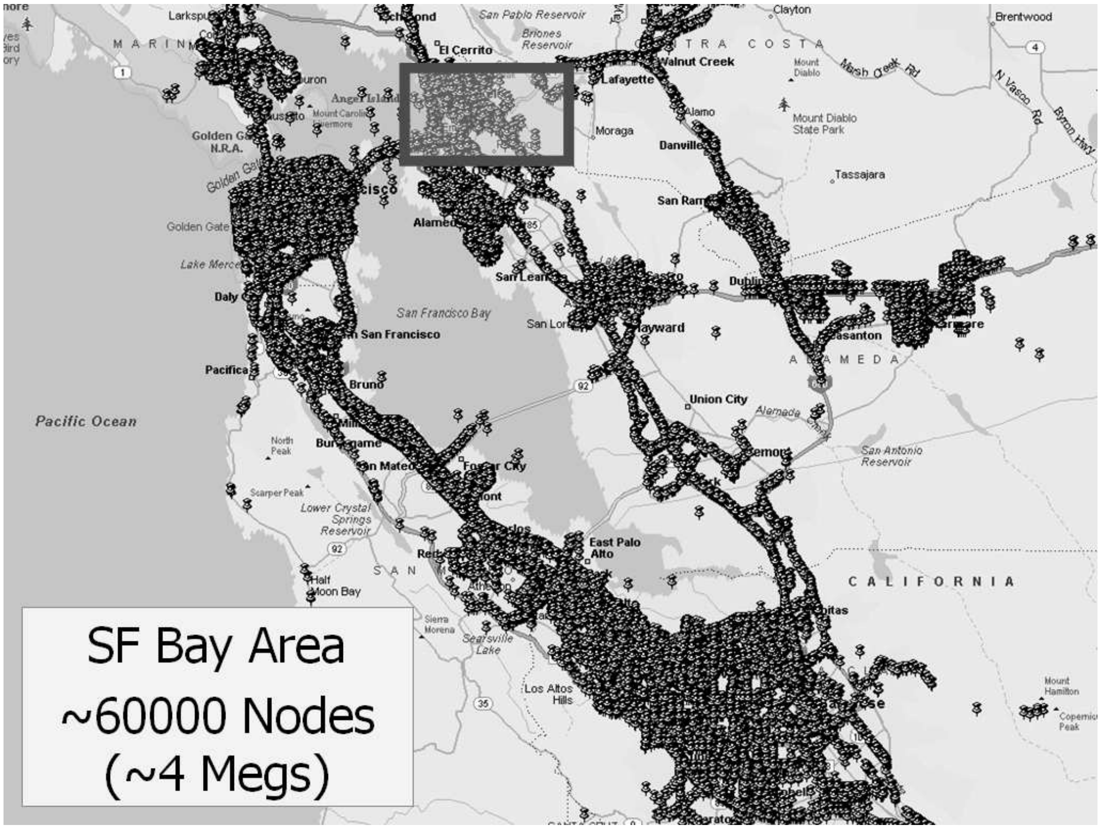


Figure 5-13. This map shows the distribution of WiFi access points in the San Francisco Bay Area as of April 2004. There are roughly 60000 known access points which takes up about 4 megabytes of storage. Over 95% of these access points were gathered from public data sources, the remaining were gathered by the author and three undergraduate students at Berkeley. The red rectangle near the top is the general location of the Berkeley campus, shown in the figure below.

Figure 5-13 and Figure 5-14 demonstrate the overall feasibility of Place Lab in terms of determining one's location and in terms of collecting the data. Figure 5-13 shows a map of access points in the San Francisco Bay Area. The coverage is fairly dense in all major urban areas, suggesting that Place Lab has strong potential in

Recently, the Special Crimes Unit of the California Deputy Attorney General's office has been actively searching the web for any publication of lists of WiFi access points in California and informing the web site administrators of their potentially illegal status. See for example http://wigle.net/phpbb/viewtopic.php?t=193 accurately determining one’s location. Furthermore, over 95% of this data was gathered by wardrivers sharing their data on wigle.net, with the remaining access points gathered by three undergraduate students at Berkeley.

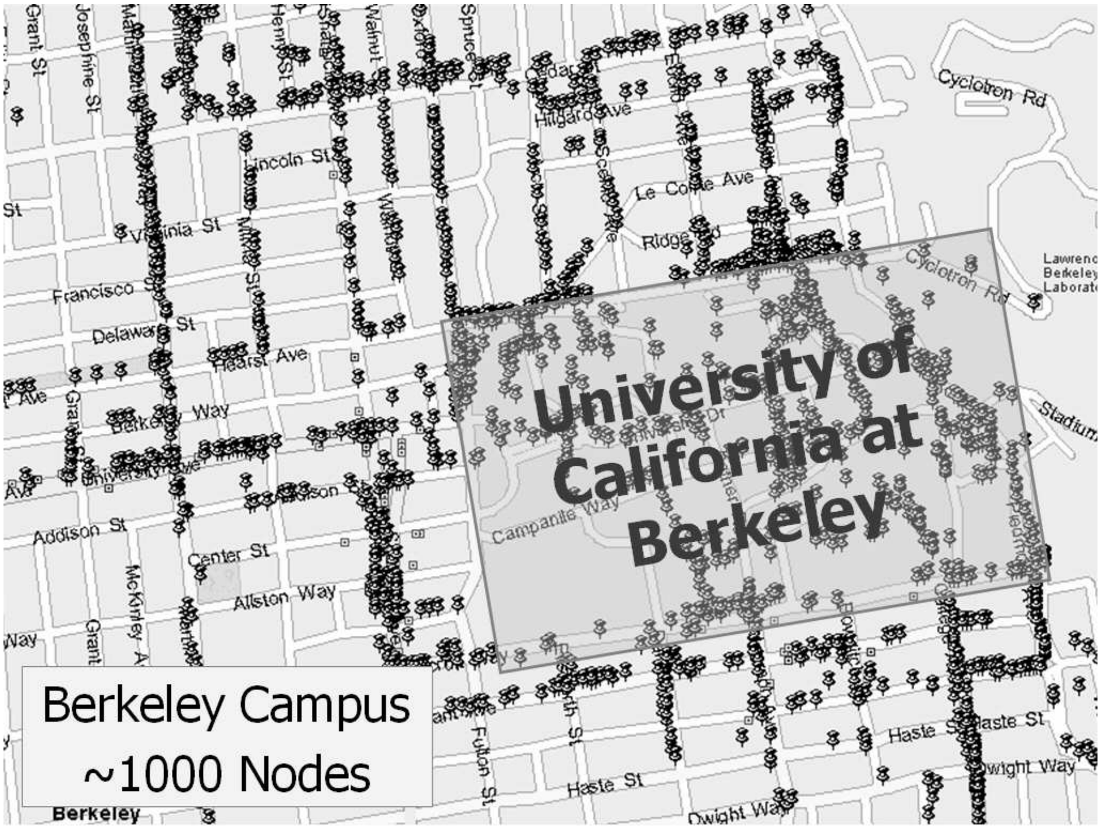


Figure 5-14. This map shows the distribution of WiFi access points around the University of California at Berkeley campus as of April 2004. There are roughly 1000 nodes here, gathered by the author and several undergraduates. This map demonstrates that Place Lab can be used to determine one’s location fairly effectively in urban areas.

Figure 5-14 shows a more detailed map of the University of California at Berkeley campus. This data shows that WiFi coverage is good in a large number of areas. It should be noted that it is likely that the empty areas represent places where the undergraduates and other wardrivers did not visit rather than places where there is no WiFi coverage.

Place Lab currently comes as part of the entire Confab source code and runtime system. Rather than implementing Place Lab directly as a sensor source within Confab, we use a level of indirection, implementing Place Lab as a GPSD server. GPSD is an open source effort to create a uniform API for accessing GPS data over a local TCP port. There are two advantages to GPSD. First, it provides a simple and programming language independent way of getting location, course, and velocity data, in a format that is easier to parse than the NMEA 0183 emitted by most GPS devices. Second, with GPSD, multiple GPS client applications can share access to GPS data without contention or loss of data. To get location data, the infospace simply runs a GPSD client that connects to the local GPSD server every 60 seconds and then adds a context tuple with the user's current latitude and longitude to that user's infospace. The advantage to using this level of indirection is that this makes it easy to replace Place Lab with alternative location sources, as long as these sources conform to the GPSD protocol.

Our current working database of WiFi access points for the San Francisco Bay Area (including the cities of San Francisco, Oakland, Berkeley, Palo Alto, and San Jose) has roughly 60000 nodes contained in about 4 megabytes of data, making it feasible to store on PDAs and laptops.

MiniGIS Operator for Providing Semantically Useful Place Names

The MiniGIS operator transforms location information from one datatype to another locally on one's computer, for example from the latitude and longitude "37.7,-122.68" to the city name "San Francisco". This is useful for two reasons. The first is because latitude and longitude are difficult to comprehend and need to be put in a format semantically meaningful to people. The second is that MiniGIS does this transformation locally without disclosing any information to equivalent network services (such as Microsoft's MapPoint[23]).

MiniGIS currently has several built-in location datatypes, including latitude and longitude, place name ("Soda Hall"), city name, ZIP code, region name ("California") and region code ("CA"), as well as country name ("United States") and country code ("USA"). MiniGIS can also be used to return the distance between two latitude and longitude pairs, as well as query for nearest locations, such as nearest places and cities.

MiniGIS uses public data sources from the USGS[24] and GeoNET[25], and has roughly 30 megabytes of data. We have also been manually collecting data for place names using a GPS system, gathering the names of local cafes, landmarks, and other points of interest.

[23] http://mappoint.msn.com

[24] The United States Geological Survey maintains a Geographic Names Information System (GNIS) for all of the states in the United States, available at http://geonames.usgs.gov/stategaz/index.html

[25] The National Geospatial-Intelligence Agency maintains the GeoNET Name Server for countries other than the United States, available at http://earth-info.nima.mil/gns/html/

5.6 Implementation

Confab is implemented in Java 2 v1.5, and is currently comprised of 550 classes and approximately 55,000 physical lines of code (not including comments and boilerplate). Confab uses HTTP for network communication and is built on top of the Tomcat web server, making extensive use of Java servlets. XPath is used as the query language for matching and retrieving XML tuples, with Jaxen as the specific XPath query engine.

The Place Lab sensor source is comprised of 10 classes and 1700 lines of code. MiniGIS is comprised of 15 classes and 3300 lines of code. Both Place Lab and MiniGIS make use of the MySQL open source database.

Confab also comes with a microphone source, which is used to estimate activity level, as well as several web-based simulators for faking location and activity data using a web browser.

5.7 Covering the Requirements

Here, we show how the design and implementation of Confab covers the requirements described in Chapters 3 and 4. Table 5-6 shows a recap of those requirements as well as what features in Confab supports those requirements. Each of the requirements is, to a greater or lesser extent, covered by the current implementation of Confab.

	Requirement	Confab Support for Requirement
End-user requirements for Ubicomp Privacy	• Clear value proposition	Service Descriptions, used by Place Bar and by Access Notifications
	• Simple and appropriate control and feedback	Place Bar, Access Notifications
	• Plausible deniability	Place Lab location ambiguity
	• Limited retention of data	Privacy tags, automatic deletion of old data
	• Decentralized control	Local processing and storage at the physical, infrastructure, and presentation layers; Place Lab location acquisition, MiniGIS
	• Special exceptions for emergencies	See BEARS emergency response service for an example (subsection 6.1.3)
Application Developer Requirements for Ubicomp Privacy	• Support for optimistic, pessimistic, and mixed-mode applications	Access Notification user interface
	• Tagging of personal information	Privacy tags
	• Mechanisms to control the access, flow, and retention of personal information (quantity)	Service Descriptions, Access Notifications
	• Mechanisms to control the precision of personal information disclosed (quality)	MiniGIS, Service Descriptions, Place Bar
	• Logging	Logging done automatically

Table 5-6. Recap of end-user requirements and application developer requirements for ubicomp privacy.

5.8 Discussion of Confab’s Architecture

In this section, we describe some of the tradeoffs inherent in Confab’s architecture.

5.8.1 Hackers

Internet users have recently been facing an escalating array of network-based attacks, including spam, spyware, phishing, and viruses. It is very likely that ubiquitous computing systems will face similar problems, and it is thus important that we try to address these vulnerabilities before these systems are widely deployed.

It is very likely that existing solutions will be updated to confront these problems in a ubicomp world. For example, one could imagine the equivalent of virus and spyware checkers for ubicomp systems. Other research ideas along these lines include a trusted computing base that could enforce certain invariants (for example, limiting the bandwidth or the amount of information disclosed to a given application), better sandboxing of applications, better ways of installing and uninstalling applications, ways of “previewing” applications without actually running them, stronger audits and better summaries to let people see where their information is going, and third-party companies that help manage users’ information for them (discussed in greater detail in Future Work in Section 8.2.7).

5.8.2 Client-based tradeoffs

Confab is structured such that information is processed locally on an end-user's computer as much as possible. As noted previously, there are interesting tradeoffs here between privacy, consistency and freshness of data, computational and storage requirements of the client, as well as overall deployability. For example, the PARCTab system [145] was designed with centralized servers, making it so that thin clients could be deployed. While there was an obvious cost in terms of privacy, the advantage here is that the incremental cost of deploying another PARCTab client is quite small. In contrast, the decentralized approach we advocate requires more powerful clients that can store a great deal of data (ex. Place Lab location data) and have enough computational power to run interesting applications. However, since this decentralized approach does not require any centralized servers, it also has the advantage of being easier to deploy, as there are fewer prerequisites needed.

The tradeoffs involved with this client-based approach are discussed in greater detail in Future Work, in Section 8.2.4.

5.8.3 Aggregation of Data

As Confab focuses more on the client-side, it does not directly address the issue of data aggregation on the server-side. However, other researchers have looked at techniques for statistically perturbing and aggregating data [9] as well as preventing re-identification of anonymized data [138]. One possibility also discussed in Future Work in Section 8.2.7 is a service that could data mine the personal information that one is disclosing to other companies, helping end-users understand what a company might be able to infer about them. For example, Acme corporation knows X and Y about you, but if you give them Z, they will also be able to infer A, B, and C as well

It is important to note that the decentralized client-based approach advocated by this dissertation is complementary to approaches that address data aggregation issues. This work looks at system architectures for how data is collected and shared with other people and services, with an emphasis on putting end-users in control. Work in data aggregation addresses privacy once the data has already been collected, emphasizing how to share interesting results from large databases of information about individuals without revealing information about any single individual.

5.8.4 Software Development Control

One major assumption here is that the software written by developers can be trusted to treat one’s personal information properly. Given the current state of software engineering, it is quite possible that there will be unintentional disclosures (for example, bugs) as well as intentional disclosures (viruses and other software exploits, as well as phishing schemes from malicious hackers). This is a generally unsolved problem in the domain of software development and beyond the scope of this dissertation. Some future work that could address this problem directly include having core Confab features as part of a trusted computing base which could enforce such things as system-wide or application-specific invariants (for example, never disclose where I was last week to any application) as well as provide a unassailable foundation for audits and summaries of disclosures. Other more general solutions include advances in proof-carrying code (code that carries quickly verifiable proofs of correctness) [109], digital signatures to verify the source of the data (so that a user knows who to complain to or who to pursue legal recourse against), white hat hackers who check the behavior of code[26], and non-technical mechanisms such as magazine reviews.

5.8.5 Proper Use of Data by Companies

The previous issue looked at whether the software could be trusted to do what it says it will do. This issue looks at whether a service provider can be trusted to do what it says it will do. For example, if a company says that it will delete one's data after a month, what guarantees are there that it will actually do so? P3P [40] provides mechanisms for communicating a company's privacy policies to end-users in a machine-readable form, but provides no way of enforcing those policies.

[26] For example, Richard Smith, a security expert, discovered that Real Networks' Real Jukebox was capturing personally identifiable information about what music people were listening to, how often the songs were being listened to, and what songs were on the hard drive) [3].

Unfortunately, short of the unrealistic solution of everyone adopting a trusted computing base, it is very likely there is no technical way of doing this. Again, the most probable solution will have to rely on market, legal, and social mechanisms for handling personal data properly [99]. Some possible ways that technology can help here include better auditing tools for companies, ways of tagging information to let applications and operators know what privacy policies or preferences there are (similar to Confab's privacy tags as described in Section 5.3 and IBM's Enterprise Privacy Authorization Language [80]), and better security tools to prevent accidental disclosures and software exploits.

5.9 Summary

In this chapter, we described the design and implementation of the Confab toolkit, focusing on the data model, the programming model, and extensions for location privacy built within Confab's framework.

Confab's data model looks at how data is represented and flows between components. This data model can be decomposed as *infospace servers*, which maintain a set of infospaces; *infospaces*, which contain a collection of context tuples that represent information about people, places, and things; and *context tuples*, which represent a single piece of information about an entity, such as their location or activity.

Confab’s programming model looks at the specifics of how a programmer would develop applications using Confab. We described several components, including operators, small pieces of reusable code designed to manage the flow of information; the Confab Client and Active Properties, two components used to interact with infospaces; Service Descriptions, which describe what information a service needs and what services it provides; and Access Notifications and the Place Bar, two graphical user interface components for helping end-users manage the flow of their personal information.

We closed with a description of extensions for location privacy built within Confab’s programming framework. This includes Place Lab, which provides support at the physical / sensor layer for acquiring location information privately, and MiniGIS, which provides infrastructure support for processing location information privately.

6 Evaluation

In this chapter, we describe the implementation of three applications we have built on top of Confab, as well as user evaluations of two of these applications[27]. The first part of this chapter looks at the range of applications possible with Confab as well as the ease of building these kinds of applications. The second part of this chapter looks at whether people could understand the privacy implications of these applications, and to a lesser extent, the privacy implications of Confab.

6.1 Applications Built Using Confab

In this section, we describe the design and implementation of three different applications we built using Confab. These include the Lemming location-enhanced instant messenger, a location-enhanced web proxy that automatically fills in location data on web forms, and the BEARS emergency response service.

6.1.1 Application #1 – Lemming Location-Enhanced Instant Messenger

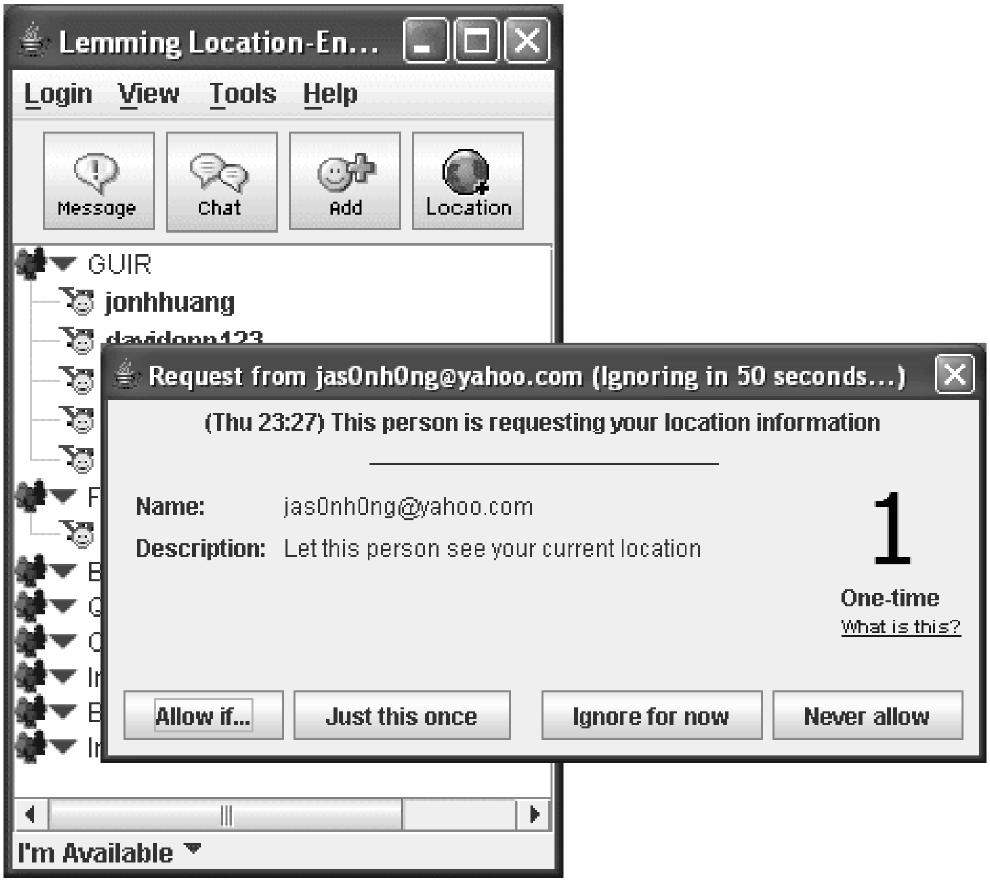
Using Confab, we have built Lemming, a new location-enhanced instant messenger client. Lemming was inspired by three observations. The first is that

[27] Parts of this chapter were previously published as [77] in The Second International Conference on Mobile Systems, Applications, and Services (Mobisys 2004)

instant messaging is used by 250 million people worldwide, growing 20% per year. What is interesting here is that these instant messenger clients are starting to move to mobile devices. Yahoo and AOL already have mobile clients available, for example. The second observation is that asking for a person's location is perhaps the most common message sent to others, especially on SMS. The third observation is that people already have a list of buddies with instant messenger, one that is often organized into meaningful categories like "work" and "friends." Rather than forcing people to create yet another account and re-create this social network, we can leverage their existing buddy list and use it as the basis for sharing location information, for example, allowing anyone in the "work" category to see location information only between 9AM and 5PM.

Lemming has two features in addition to standard instant messenger clients. The first new feature is the ability to request a user's current location (see Figure 6-1). When a location request is received, the end-user gets an access notification. The end-user can choose "Never allow" to never allow the requestor to see her location, "Ignore for now" to ignore this current request (the default), "Just this once" to allow the request just this once, or "Allow if…" to always allow requests under certain conditions, such as from 9AM to 5PM or only between Monday and Friday.

From a software architecture perspective, when a location request is received, the end-user's instant messenger client issues a query to her infospace for her current location. Currently, Confab does not provide mechanisms for authentication, relying instead on the application itself to manage it. The infospace checks if there is a context tuple representing location information, and then checks the age of the tuple to see if it can be considered "current" (by default, this is set to twenty minutes).



Request from jas0nh0ng@yahoo.com (Ignoring in 50 seconds...)

(Thu 23:27) This person is requesting your location information

Name: jas0nh0ng@yahoo.com

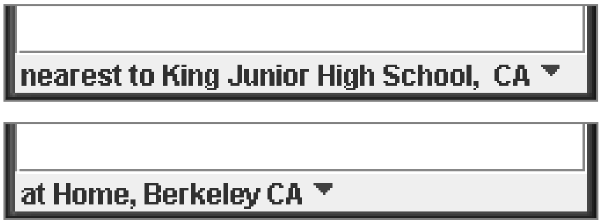
Description: Let this person see your current location

1

One-time

What is this?

Figure 6-1. Lemming is a location-enhanced messenger that lets users query each other for their current location information. This screenshot shows the UI that lets a requestee choose whether or not to disclose their current location. The large "1" on the side represents that this is a one-time disclosure rather than a continuous disclosure of location information.



nearest to King Junior High School, CA ▼

at Home, Berkeley CA ▼

Figure 6-2. This location-enhanced messenger lets users set an away message describing their current location, which automatically updates as they move around.

At this point, the tuple flows through the out-operators defined in the infospace. The three operators of interest here are the Enforce Access Policies, Interactive, and MiniGIS operators. The Enforce Access Policies operator checks if there is an existing policy associated with the requestor and applies that policy if it exists. The Interactive operator also checks if there is an existing policy, and if there is not, brings up the user interface shown in Figure 6-1, letting end-users set a policy. Lastly, the MiniGIS operator runs, transforming the data from “latitude and longitude” into “place.”

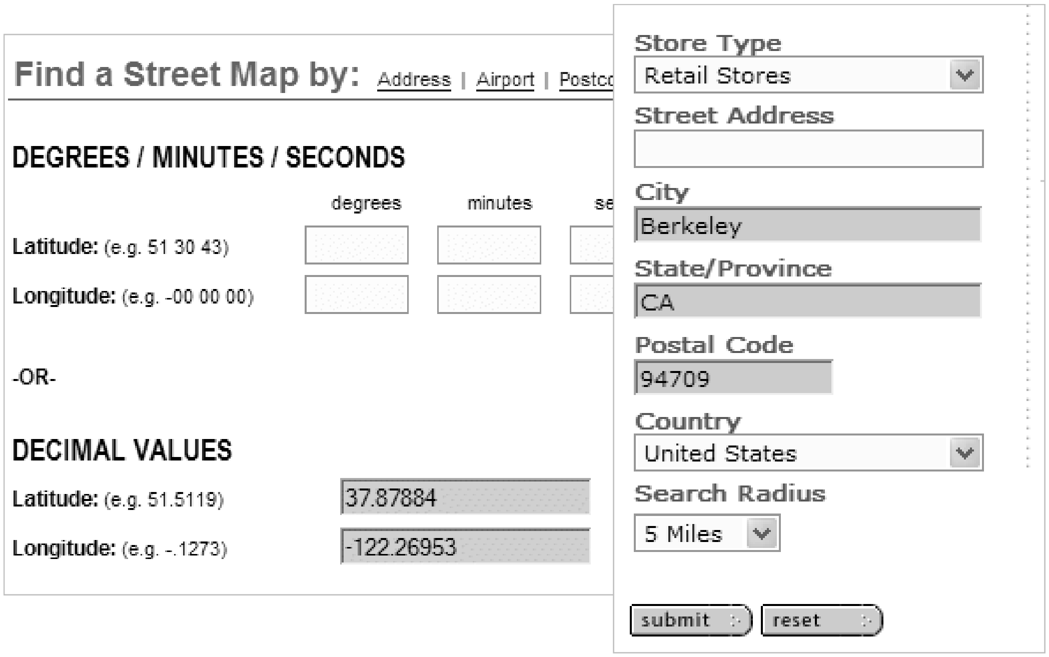
The second new feature is the ability to automatically display one’s current location as an away message that automatically updates itself as one’s location changes (see Figure 6-2). The Lemming instant messenger client sets up a query to get the nearest “place” every 60 seconds, and then displays this place as the away message. Lemming currently defines three place descriptions based on the user’s distance to that place: “at”, if the distance is less than 10 meters; “near”, if the distance is less than 100 meters; and “nearest to” otherwise.

Lemming uses the Hamsam library for cross-platform instant messaging[28]. Lemming is roughly 2500 lines of code across 23 classes. It took about 5 weeks to build, with the majority of the effort and code devoted to the GUI. Here, Confab provides support for acquiring location information, storing location information and privacy preferences, making location queries, automatically updating location information for the away message, and processing location information using MiniGIS.

6.1.2 Application #2 – Location-Enhanced Web Proxy

We have built a location-enhanced web proxy that provides two features. The first is that it can automatically fill in fields on existing web sites (see Figure 6-3 and Figure 6-4). The second is that it enables the creation of location-enhanced web sites.

[28] http://hamsam.sourceforge.net/



Find a Street Map by: Address | Airport | Postc

DEGREES / MINUTES / SECONDS

degrees minutes se

Latitude: (e.g. 51 30 43)

Longitude: (e.g. -00 00 00)

-OR-

DECIMAL VALUES

Latitude: (e.g. 51.5119) 37.87884

Longitude: (e.g. -.1273) -122.26953

Store Type: Retail Stores

Street Address:

City: Berkeley

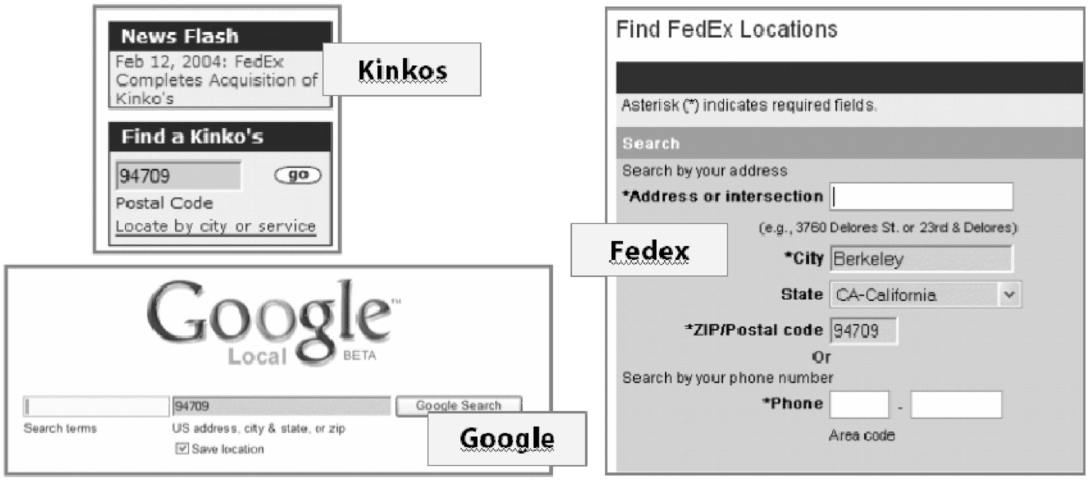
State/Province: CA

Postal Code: 94709

Country: United States

Search Radius: 5 Miles

Figure 6-3. The location-enhanced web proxy automatically fills in location fields on web pages. The page on the left is from MapQuest (http://mapquest.com), with latitude and longitude automatically filled in. The page on the right is a store finder from StarBucks (http://starbucks.com), with city, state/province, and postal code automatically filled in.



News Flash

Feb 12, 2004: FedEx Completes Acquisition of Kinko's

Find a Kinko's

94709 go

Postal Code

Locate by city or service

Google Local BETA

Search terms

US address, city & state, or zip: 94709

☑ Save location

Find FedEx Locations

Asterisk (*) indicates required fields.

Search

Search by your address

*Address or intersection

(e.g., 3760 Delores St. or 23rd & Delores)

*City Berkeley

State CA-California

*ZIP/Postal code 94709

Or

Search by your phone number

*Phone -

Area code

Figure 6-4. Some more examples of automatically filling in current location information.

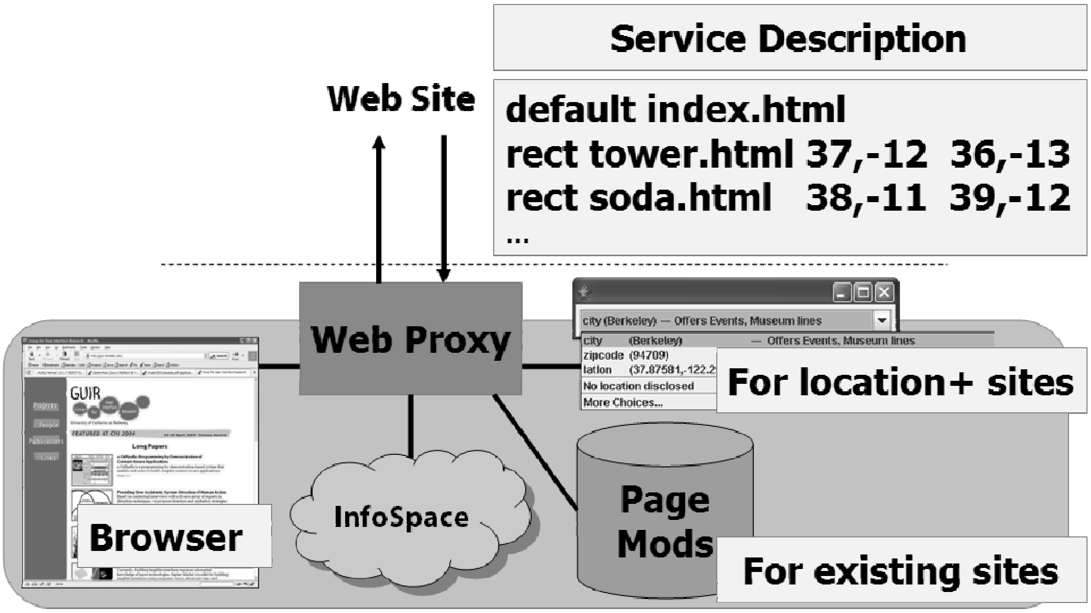


Figure 6-5. System architecture for the location-enhanced web proxy. The web proxy has two features. The first is to automatically fill in location information on existing web sites. It does this by inserting location information on pre-defined URLs (see

Figure 6-6). The second is the creation of location-enhanced web sites. When a web site is first accessed, the proxy checks if there is a service description associated with it (in the same manner web crawlers look for robots.txt). If there is one, the proxy automatically downloads it and configures the Place Bar. When a page is requested, the proxy automatically puts in the appropriate location information in the HTTP protocol. The web server can then return a different page depending on the person's location.

The system architecture for the location-enhanced web proxy is shown in Figure 6-5. To automatically add in location information on existing web sites, on start up, the web proxy loads up a configuration file that describes which URLs to look for, which HTML input fields to modify, and what values to insert in those fields (see

Figure 6-6). Some possible values include one's current city, state, ZIP code, and latitude and longitude. Users can run this proxy locally on their computer and set their web browser to use this proxy. Whenever the proxy detects one of the predefined URLs, it modifies the HTML, inserting the current location information and highlighting the modified fields in blue.

```
PageModification
URL=http://www.mapquest.com/
URL=http://www.mapquest.com/index.html
state=RegionCode
zipcode=ZIPCode
city=CityName

PageModification
URL      = http://www.starbucks.com
txtCity  = CityName
txtState = RegionCode
txtZip   = ZIPCode
```

Figure 6-6. An example configuration file that specifies what URL to look for, what HTML field on that page to modify, and what value to insert. In this case, the first example means to look for the Krispy Kreme URL, and if it is found, then find the HTML fields named "aCity", "aState", and "aZip", inserting the current location values for "CityName", "RegionCode", and "ZIPCode" (where "CityName", "RegionCode", and "ZIPCode" are predefined keywords).

To enable location-enhanced web sites, the web proxy automatically checks for a service description file when it first encounters a web site. This service description is the same as described in Figure 5-6, and is expected to be located in a well-known location (currently, this is `/service.xml`). This service description file is transparently downloaded by the proxy, and then used to configure the Place Bar. Whenever a web page is requested, the web proxy inserts the appropriate location information in the HTTP protocol. The example illustrated in Figure 6-5 shows a server-side location configuration file that returns the file `tower.html` if the user is in the rectangle bounded by the points (37,-12) and (36,-13).

The upshot of all this is that web servers can now return different content depending on the location of the requestor. As proof of concept, we have created a simple Java servlet that reads in a location configuration file (see top-right of Figure 6-5) that does exactly this, allowing a web designer to define different location regions and what web pages should be returned if the requestor says she is in that region. To facilitate parsing and to simplify the learning curve, this location configuration file has the same file format as HTML image maps.

To help protect the privacy of the user, the proxy is restricted to accept connections only from localhost. The location-enhanced web proxy is roughly 800 lines of code, added to an existing base of 800 lines of code from an open-source web proxy. It took about one week to build. Here, Confab provides support for making location queries for one's current location, automatically updating one's location, as well as processing location information using MiniGIS.

6.1.3 Application #3 – BEARS Emergency Response Service

One emerging application for location-enhanced phones is Enhanced 911. E911 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.

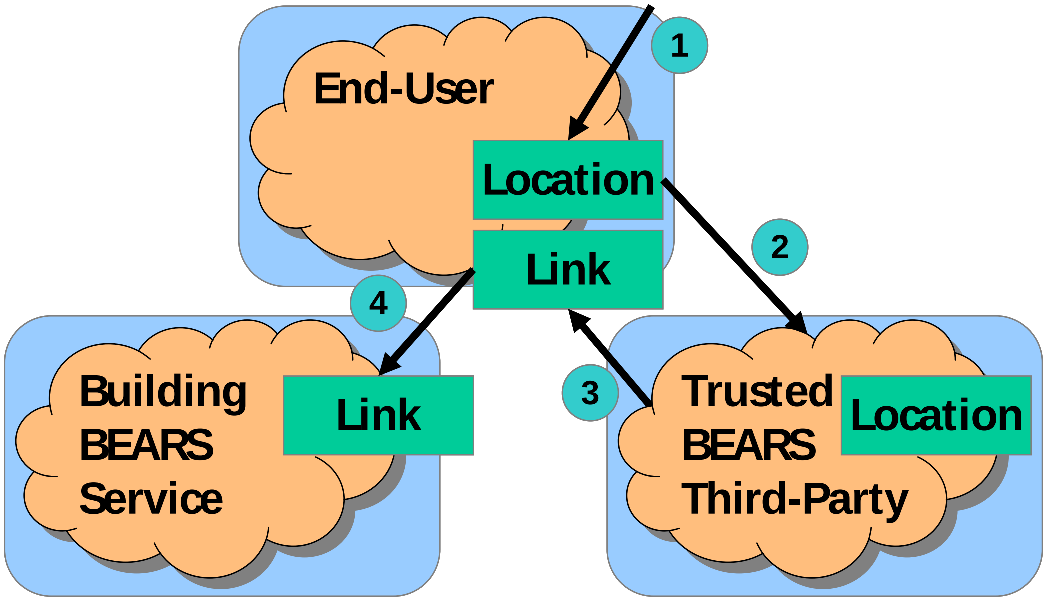


Figure 6-7. An example setup of the BEARS emergency response service. First, an end-user obtains their location (1) and shares it with a trusted third-party (2). The end-user gets a link (3) that can be sent to others, in this case to a building (4). If there is an emergency, responders can traverse 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).

BEARS is a system we are developing to handle these cases.[29] 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, emergencies are rather rare, and

[29] BEARS stands for Berkeley Emergency Action Response Service.

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 in case of emergencies. 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 [2]. These services would have a significant market incentive to use location information only for stated purposes and possibly a legal obligation as well.

Figure 6-7 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, an end-user obtains his location. He periodically sends his location to the trusted third party, which gives him one or more named links back to this data. The end-user 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 from the third party, with the third party also notifying end-users that their information has been discloesd. This approach allows emergency responders to get critical location information, provides a level of redundancy should the user's device or location systems fail or if the end-user is incapacitated, and provides a basic level of privacy.

The BEARS client is roughly 200 lines of code and took about 2 days to create. The reason for its small size is that there is no GUI. Here, Confab provides support for making continuous location queries, as well as making updates to both the trusted third-party and to the building server.

6.1.4 Other Applications Built Using Confab

We have also used Confab to build prototypes of applications that have minimal privacy concerns. One that is currently in progress is emergency response support to help firefighters on scene. Our prototype uses sensors and PDAs to automatically gather and disseminate information about the fire and about firefighters to other nearby firefighters [83]. This prototype was developed by one graduate student and two undergraduate students.

Another is the liquid distributed querying system for supporting database operations, such as join or project, for streaming data and across multiple infospaces [74]. This system was developed by three graduate students.

6.2 User Evaluations

We ran informal task-based user studies with nine people to understand how well they could understand the Lemming location-enhanced messenger, as well as a web-based mobile tour guide application that made use of the location-enhanced web proxy. We asked participants to go through several tasks. With Lemming, we asked them to request someone's location information, respond to someone's location request, and set their away message to show their current city location. With the mobile tour guide, we asked them to see what events were happening in that city, and to find several points of interest currently near them.

Our participants were very familiar with using the web and instant messenger, but none had extensive programming experience. We were looking for three things in our studies. First, we wanted to see if our participants could accomplish basic tasks correctly. This included being able to understand the choices presented to them by the user interfaces (specifically, the access notifications and the Place Bar), as well as if they could use these user interfaces to make desired choices.

The second thing we wanted to understand is the general conceptual model that people had from using these applications. It is important that the system work roughly the way users believe it does, because, as Norman [111] has pointed out, a mismatch in conceptual models can lead to slow performance, errors, frustration, and in this case, undesired privacy violations.

The third thing we wanted to understand is if our participants still had serious privacy concerns. Would they want to use it? What barriers to entry remained?

While it would have been possible to collect performance metrics, we were more interested in getting information about the qualitative user experience and perceptions of privacy rather than bottom-line data. Below, we describe the results of our informal user studies, grouping the feedback into three categories: general user feedback, feedback about the access notifications, and feedback about the Place Bar.

6.2.1 General User Feedback

All of our participants were able to successfully complete the basic tasks involved with these two applications. For the most part, our participants did understand the user interfaces, though participants had some difficulties with the Place Bar (described in more detail below).

Interestingly, in our user studies, we did not say how the location information was acquired, but all nine of our participants assumed that their location information started with them, with no third parties involved. Although this is preliminary evidence, it suggests that this may be the default conceptual model for non-badge systems. Anecdotally, we have observed this to be the case for many location-enhanced phones as well. The implication here is that Confab supports peoples' default conceptual model, as location information does start with the end-users.

Participants also did not have any marked concerns with respect to privacy. Our participants felt that they could easily make disclosures they were comfortable with. We believe this is the case because the two applications fit well in our participants' existing comfort zone. Our participants were already very familiar with instant messaging and web browsing, and so these applications only added a small feature, asking them to take a small step forward, rather than requiring a large shift in work practices. Furthermore, these applications were designed to provide value in an obvious manner to end-users. These two points are in contrast to nurses using locator badges, in that the locator badges asked nurses to make a fairly large change in how they did things, and did not provide immediate value to the wearers of those badges. Although this issue still bears further investigation, it seems that the deployment of new ubicomp technologies can be smoothed out by focusing on these two points.

Participants also seemed quite enthusiastic about new possibilities. Many of them suggested new applications, such as the ability to check for the length of movie lines and bus lines, as well as the ability to make sure that children were safe.

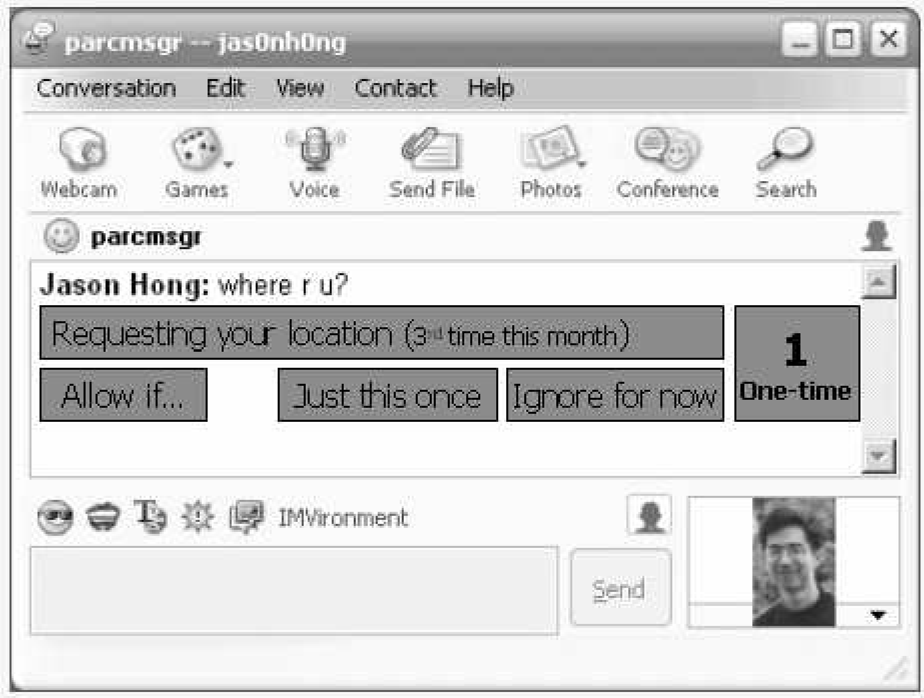
6.2.2 User Feedback on Access Notifications

Broadly speaking, all of the participants in the user study understood the access notifications and could make their desired choice. When shown the access notification user interface, everyone chose “Just for now.” There were only two minor points of confusion. The first point of confusion was what “one-time” disclosures meant, but everyone quickly understood it once they saw the small popup describing the difference between one-time and continuous disclosures. The second point of confusion dealt with what a buddy would see if the “Ignore for now” button was selected. It was not immediately clear if a buddy would see something alone the lines of “you are being ignored” or “UNKNOWN.” As described earlier, both Confab and Lemming use “UNKNOWN” to give some level of plausible deniability.

Participants also had several suggestions for improving the access notification user interface. Most of the participants thought there still too much text. Many participants also suggested using location as a potential option for limiting who could see what and when. Once they understood that the computer could determine its location, many of them suggested using places such as “work” and “home” to control access to their location information.

Based on this feedback and from comments from other researchers, we have created a mockup of a revised user interface for access notifications (see Figure 6-8). Based on the latest version of Yahoo Instant Messenger as of this writing, this design has a tighter integration with the instant messenger client, displaying the notification within the message window rather than it being a separate dialog box. It also reduces the amount of text shown (a common criticism from users), and reduces the number of buttons from four to three, putting the “Never allow” option under the “Allow if…” button. The design rationale here is that “Never allow” is not used as often as “Just this once” and “Ignore for now”, and so putting the “Never allow” button under “Allow if…” could reduce potential clutter.

This design has the advantage that it integrates instant messaging and location requests better. Location requests can go into the same window that messages go, so that users do not have to learn another user interface or deal with another popup window. These instant messaging windows are also persistent, letting people easily see who has made any recent requests.



Jason Hong: where r u?

Requesting your location (3rd time this month)

Figure 6-8. Mockup of the revised access notification user interface. A request for one's location is integrated more tightly with the instant messenger client. The amount of text is reduced, and there is also a small note stating how many times this person has requested the information.

There are four features that this mockup does not show. The first is the ability to set preferences for an entire group, for example, always allowing friends to see location information, rather than just a single individual. The second is setting preferences based on location, for example, allowing any buddy in your "work" group to see your location if you are in the office. The third is a preview for what location information will be revealed. The fourth is the capability to do white lies, saying you are in one place when you are really somewhere else. We have left the design of these features as future work.

Participants also suggested several extensions to Lemming, such as making it easy to bring up maps showing someone's location, and the ability to do continuous queries when out with a group to make sure no one is being left behind.

6.2.3 User Feedback on Place Bar

In general, our participants understood the basic idea of the Place Bar, but found that it put too many concepts in a single user interface widget. Simply providing a text-based list of what features were or were not available made sense at one level, but did not make it clear where that feature was on a web site, and whether it was worth having or not. It was also not immediately clear to some participants that the service levels were inclusive, in that choosing a finer-grained level of location also included coarser-grained ones as well. For example, choosing to disclose street level would provide the services for street level, ZIP code, as well as city.

Participants also noted that the Place Bar had some confusing terminology, for example "latlon" for "latitude and longitude." Participants also questioned whether it was necessary to show latitude and longitude at all since few people have an intuitive understanding of it. For example, it is not obvious that 37° 46′ N, 122° 26′ W is San Francisco. Participants also suggested that rather than simply showing one's current location, the Place Bar would be useful in also handling pre-defined static locations as well, such as “home” and “work.”

Overall, the basic idea of the Place Bar, a reusable user interface widget for seeing and controlling what level of information goes out for push transactions, seems to be correct. However, participant feedback strongly suggests that more work needs to be done to make it simpler to understand and more effective in practice.

6.2.4 Summary of User Studies

To recap, we were looking for three things in our user studies:

- Can participants accomplish basic tasks correctly, understand the choices presented to them, and make disclosures they were comfortable with?
- What conceptual model did participants have?
- Did participants still have any privacy concerns that would cause them not to want to use these applications?

With respect to the first point, our participants could accomplish all of the basic tasks, and did understand the choices presented to them, but also had a great deal of feedback on how to improve the access notification and Place Bar user interfaces. With respect to the second point, all of our participants seemed to have the same conceptual model, namely that their location information starts with them. We see this as a positive aspect of Confab, because Confab matches this conceptual model.

With respect to the third point, participants seemed fairly enthusiastic about these applications. We believe this is the case here because our applications were only a small step forward from activities they were already familiar with, and provided immediate value to them (rather than providing value to others first, as was the case with the nurse locator badges).

6.3 Summary

In this chapter, we described the evaluation of the Confab toolkit. This evaluation is comprised of two parts. The first part looked at what kinds of applications can be built on top of Confab and the level of difficulty in doing so. We did this by building three different applications, including a location-enhanced instant messenger, a location-enhanced web proxy, and an emergency-response application.

The second part of this evaluation looked at the utility and usability of these applications, focusing primarily on the access notification and Place Bar user interfaces. Based on participant feedback from informal user studies with nine people, we have developed mockups fixing problems and adding new features to these user interfaces.

7 Related Work

In this chapter, we provide an overview of related work and show how our work differs or builds on previous work.[30] The related work is grouped together into several sections, looking at support for building ubiquitous computing systems, digital rights management, and support for anonymity.

7.1 Support for Building Ubiquitous Computing Systems

There has been a great deal of work at providing programming support for various aspects of ubiquitous context-aware computing. This includes the PARCTab system [129], Limbo [43], Sentient Computing [8], Cooltown [87], the Context Toolkit [45], Contextors [41], SpeakEasy [48], XWeb [113], one.world [65], MUSE [33], Solar [35], Gaia [126], iRoom [85], and Stick-E notes [118]. Each of these systems supports the development of different kinds of ubiquitous computing applications. At a high level, Confab builds on all of this previous work, with the key difference being that Confab's architecture and mechanisms are focused on helping application developers and end-users manage personal privacy. We describe each of these systems and how Confab differs below.

[30] Parts of this chapter were previously published as [77] in The Second International Conference on Mobile Systems, Applications, and Services (Mobisys 2004)

The PARCTab system [129], Limbo [43], and Sentient Computing [8] are three centralized systems that introduce different infrastructural approaches to ubiquitous computing. Both the PARCTab system and the Sentient Computing project used a centralized store for contextual information about people, places, and things, for example, the name and location of a person. Limbo used a centralized tuple space to coordinate mobile computing applications. These centralized data stores separate components in time and space. These centralized data stores also represent a way of coordinating components without them needing explicit knowledge of one another. For example, an application does not need to know how the location information for a person was acquired, just whether that data is available or not. To check if that information is available, the application simply needs to query the centralized data store. For privacy reasons, rather than using a centralized store for such information, Confab uses decentralized stores, with each person storing contextual data about themselves on their personal devices.

In many ways, Confab's data model can be thought of as a logical evolution of the PARCTab's Dynamic Environments. Dynamic Environments are the centralized data stores in the PARCTab system, and are associated with relatively large places such as buildings. Each Dynamic Environment contains personal information about people, places, and things within its purview. As people move from place to place, they also switch which Dynamic Environment they are using. The key differences Confab makes are decentralization of data so that personal information is stored and processed on the end-user’s computer as much as possible, smaller granularity infospaces that represent individuals rather than large areas and the people within them, a greater range of mechanisms for privacy in both the data model and in the programming model, and compartmentalized extensibility thru operators.

Cooltown [87] looked at extending web technologies to the physical world, providing web-based points of presence for entities such as people, places, and things. Associations between physical entities and virtual web pages were established using beacons wirelessly sending out URLs or scannable codes such as barcodes. Cooltown also provided data transfer between entities, so a person could send a file from their PDA to a projector to have that projector display a presentation. Like Cooltown, in Confab, entities can have data associated with them (via an infospace). Similarly, Confab leverages many of the ideas embodied in the web. However, rather than presenting HTML content for people, Confab uses XML content for processing by computers. Confab also focuses less on data transfer issues and smart spaces, and more on personal services that can be run on single-user mobile devices.

The Context Toolkit [45] and Contextors [41] both used a modular approach for acquiring, processing, and refining contextual information from sensors. With the Context Toolkit, the primary abstractions were context *widgets* for acquiring sensor data (e.g., a location widget might have GPS or an Active Badge as the underlying location source), *interpreters* that refine or re-map low-level contextual information (e.g., using GPS latitude and longitude data to determine one's zip code), and *service discovery* for seeing what widgets are running on a given computer. Contextors pushed this idea of a context widget even further, adding more capabilities and control over how sensor data was acquired and processed.

Confab focuses less on reusable components for acquiring and processing data, putting the focus more on the data format used. The issue here is that these components may or may not be able to run on arbitrary devices, due to programming language or runtime constraints. For example, Java cannot run on many small devices. Furthermore, Confab emphasizes a more disciplined data flow for privacy reasons, pushing as much of acquisition and processing as possible onto local devices.

Both SpeakEasy [48] and XWeb [113] look at connecting arbitrary devices together. SpeakEasy proposed standard meta-interfaces for all components (e.g., file systems, projectors, phones, PDAs, laptops, DVD players, TVs). These meta-interfaces describe what the component is and what data types it supports. End-users can then use a software program to discover what components are available and connect arbitrary components together, for example, "this movie file" to "that TV" (all components are implicitly network-enabled). If the destination understands how to process the source data type, then the destination processes the data normally. In the example above, the TV would simply play the movie file. If it does not, then the source is expected to provide mobile code which can be run on the destination, allowing the destination to process that data type. In the example above, the TV would request and download mobile code from the movie file (or from a web site that brokers these kinds of requests), and then be able to show the movie. The advantage of this approach is that it manages the interoperability issue, in that every component does not need to understand how to interoperate with every other component a priori. In other words, devices do not have to be manually upgraded with new software, and old devices will be able to use new services as they are developed without having to load new software. XWeb proposed a simple protocol and data format for connecting arbitrary devices together, in a way that can be considered a logical extension of HTTP for devices. Essentially, every device exposes an XML tree describing itself. For example, a light switch might expose its name and whether it is on or off. Another device that speaks the XWeb protocol can send an XWeb request to change the state of the light switch to off. It should be noted that XWeb does not specify how the XML tree and the underlying physical resource are linked (e.g., turning the light switch on would update the XML tree, and setting the state to "on" in the xml tree would turn the light switch on), just that they are linked somehow. Confab is focused less on these connection issues and more on representing contextual information about people, places, and things, the privacy issues involved in doing so, and constructing useful services on top.

one.world [65] proposed a programming framework and system architecture for developing highly dynamic computing environments. It provides several programming abstractions to simplify common operations in these kinds of environments, such as service discovery, checkpointing, and migration. Confab also provides a programming framework and system architecture, though Confab is implicitly targeting an alternative design space, primarily mobile applications with an emphasis on end-user privacy and control over the flow of their personal information.

MUSE [33] is a service-oriented system for sensor-based systems. Built on top of Sun Microsystem's Jini framework, the focus was on providing a uniform service architecture for accessing sensor data as well as managing the ambiguity inherent in sensor-based systems using Bayesian reasoning. While Confab shares many of the same goals, our focus was more on personal services using information about end-users rather than implicitly supporting public services, such as sensor data about rainforests or traffic information.

Solar [35] is a platform for context-aware mobile applications. It supports context collection, aggregation, and dissemination, through the dynamic composition of a graph of operators to compute desired context from appropriate sources. To optimize scalability, parallelism, and load balancing, Solar is designed such that these operators are placed onto the infrastructure. The reasoning here is that by offloading computation and storage onto the infrastructure, thinner mobile clients can be deployed. In contrast, for reasons of privacy, we take the opposite approach with Confab, placing as much of the context collection, aggregation, and dissemination on a single device and providing better user interfaces for managing the flow of information. It should be noted that Solar is conceptually similar to the liquid distributed querying system [74] developed on top of Confab, where a single query can be distributed across multiple infospaces to aggregate and disseminate contextual information.

Both Gaia [126] and iRoom [85] provide system support for smart spaces, electronically augmented physical rooms. Both provide abstractions and basic services to help application developers construct programs for use in these kinds of interactive workspaces. The key difference here is that Confab is designed more for mobile devices and applications, with an emphasis on privacy. Confab builds on several ideas embodied by the iRoom software suite. Central to the iRoom is the EventHeap, a shared tuplespace for the room in which input devices can place events and output devices can receive events. This level of indirection encourages looser coupling between application components and fosters greater overall robustness. Confab uses a similar approach with its infospaces, separating sources of data (such as sensors) from the services and applications that use them, with little or no knowledge of each other. The main difference between the EventHeap and Confab is that Confab is specialized for building privacy-sensitive systems. Confab also looks at supporting multiple infospaces to represent people, places, and things, rather than just one tuplespace to represent all events and information within a place. Again, Confab takes a decentralized approach, placing information about end-users on their computers as much as possible. Lastly, the EventHeap is highly tuned for multiple concurrent processes adding and removing tuple data, whereas we have not done this for Confab at this time.

Stick-E notes [118] was a novel approach to rapid authoring of location-aware content. Rather than developing applications by constructing programs in the traditional sense, applications could be created by authoring content and then specifying when and where that content should be displayed. As an analogy, it is the same difference between a general purpose programming language and specialized content languages like HTML. General purpose programming languages provide a great deal of flexibility, but often at the cost of complexity, making them somewhat difficult to learn. In contrast, a content language makes it easy to do certain kinds of tasks, such as handling the layout and presentation of information, but hard or impossible to do tasks that it was not designed for, such as animations or network security. With Confab, our focus was to provide flexibility with respect to privacy, though it would be interesting to explore what kinds of specialized content languages would be possible to simplify the creation of a larger set of ubiquitous computing applications. We discuss this possibility in Future Work section 8.2.6, Tools for Facilitating the Creation of Ubicomp Applications.

Confab also builds on the work by Spreitzer and Theimer [137], who describe an architecture for providing location information. In their architecture, each user owns a User Agent that collects and controls all personal information pertaining to its user, and any request for such information must be routed through the User Agent which enforces predetermined access policies. Confab takes this same basic approach and extends it with a wider range of privacy mechanisms, including notifications, tags, logging, and interactive requests, to support the development of pessimistic, optimistic, and mixed-initiative type applications.

7.2 Digital Rights Management for Privacy

There has also been some previous work on using digital rights management in managing personal information. Langheinrich [92] described pawS, a privacy awareness system for ubicomp that lets deployed systems announce P3P policies through beacons, describing what data is being collected. The pawS system also offers database support for enforcing those policies. The difficulty here is that pawS requires cooperation and mutual respect from all parties involved, including the end-users that are being tracked and the systems that are tracking them. With Confab, we designed the system to acquire and process as much data locally as possible, so that for certain kinds of applications (namely those where data can be cached and do not require network interaction), no such cooperation is required. This approach by itself, however, has obvious limits. Towards this end, we have included in Confab the notion of privacy tags, which are similar in spirit to the database enforcement policies in pawS.

Similarly, IBM has also introduced an Enterprise Privacy Authorization Language (EPAL) [80] that lets developers describe privacy policies and attach those privacy policies to data as it flows through a corporation. The privacy tags in Confab are similar in spirit to these ideas, focusing instead on privacy for individuals rather than data management within a corporation. Confab's privacy tags also introduce further digital rights management ideas, such as using location as a parameter for digital rights management, enforcing a maximum number of past sightings, and peer enforcement of tags. For example, one thing that Confab can support that IBM's EPAL cannot is automatic deletion of old data based when a device has moved outside of a given location.

7.3 Support for Anonymity

There has been a great deal of work in providing levels of anonymity in networked systems. One system of note here is Gruteser and Grunwald's work on spatial and temporal cloaking [70], in which a trusted proxy is used to adjust the resolution of location reported to services based on the density of users in a region. Since many users report their location through the proxy, user density is known. Thus, the proxy can provide k-anonymity, that is hiding one's precise location by returning an area that has k-1 other people. Sweeney [138] has proposed a general approach for doing k-anonymity for static database tables, aggregating data together into buckets to reduce identifiability. Another approach is to use mixes to make it harder to do traffic analysis (e.g., [23]). With mixes, the basic idea is to route data across several well-known mix servers, with the data being encrypted and hidden within other traffic as it is sent to another mix server. The advantage to this approach is that it is difficult to trace where a packet came from, where it is going, who sent it, and when it was sent. An attacker must compromise all of the mix servers to acquire all of this information (though an attacker could, for example, compromise the initial mix server to see where a packet came from, or the final one to see where a packet is going). The disadvantage to this approach is that routing data across these mixes adds a fair amount of latency to network traffic, and only provides anonymity. As we noted in the first chapter, while anonymity has its uses, in many cases it often does not provide a useful level of privacy when communicating with people in one's social network.

Confab currently does not have any built-in support for managing anonymity or for defeating traffic analysis, but rather could rely on existing techniques for doing so, including onion routing [34], where packets are encrypted and sent back and forth between multiple routers to make it difficult to analyze traffic, and dog leg routing [136], where packets are sent to a well-known home address that always knows where a mobile node is and can forward the packet accordingly.

Confab also provides support for applications in which anonymity is not always useful, as in some situations with family, friends, co-workers, and certain paid services. For example, with family, friends, and co-workers, if they are requesting your current location, they already know your identity, and hence anonymity is not useful. With paid services, you need a way of paying for those services, and anonymous electronic cash has not been widely deployed. While in theory one could construct an anonymous paid service, it does not seem practical to do so at this point in time.

7.4 Summary

In summary, while there have been many toolkits and infrastructures providing programming support and abstractions for sensors, and while there have been many individual techniques for managing privacy, Confab focuses on providing an extendable design that provides software architecture support for building privacy-sensitive ubicomp applications. Confab provides reusable mechanisms for end-users in managing personal information, as well as mechanisms and abstractions for application developers designing privacy-sensitive ubicomp systems.

8 Future Work

In this chapter, we outline several directions for future work in the field of privacy-sensitive ubiquitous computing. We have divided this chapter into two parts, looking at short-term future work and long-term future work.

8.1 Short-Term Future Work

In this section, we look at areas for future work in the short term, including continued development and evaluation of ubicomp applications, better integration of access notifications with instant messengers, developing alternative user interfaces to the place bar, and implementation of peer enforcement of privacy tags.

8.1.1 Continued Development and Evaluation of Ubicomp Applications

This dissertation presented an informal evaluation of Confab, in terms of support for building privacy-sensitive applications and user studies of those applications. However, this only represents a first step towards increasing our understanding of how to build and deploy privacy-sensitive ubiquitous computing applications. Further evaluation that provides stronger evidence that this approach simplifies the process of creating privacy-sensitive ubicomp applications is still needed, in terms of providing better and understandable privacy and in terms of streamlining the development of ubicomp applications in general.

We have already started one form of assessment along these lines, in terms of making the source code freely available.[31] Other ways of assessing Confab in the future include doing formal usability studies on the toolkit to get feedback from application developers, and deploying real applications to see how people use them in realistic situations.

Another path along these lines is to re-work Confab so that it can run on PDAs and cell phones. As noted in section 5.3, a simplified and reduced functionality version of Confab has already been implemented in C++, our goal here is to create a more fully-featured and robust version. The main implementation of Confab currently runs as a background service on relatively high-end computing devices like laptops. While this is reasonable as a proof of concept, it also limits the kinds of applications that can be implemented and deployed. Re-implementing the core ideas in Confab for smaller clients would allow us to explore a richer design and interaction space, and would also push the privacy and ubicomp aspects of this research even further.

[31] Confab can be downloaded at http://sourceforge.net/project/confab

8.1.2 Better Integration of Access Notifications with Instant Messengers

As described in Section 6.2.2, we believe that the access notification user interface can be more tightly integrated with existing instant messenger clients. The advantage of this approach is that it reduces the number of user interfaces that end-users have to learn, and also reduces the number of places end-users have to go to in order to check their how their personal information is being used.

8.1.3 Develop Alternative User Interfaces to the Place Bar

As described in Section 6.2.3, the Place Bar did not work as well as we had hoped. Participants understood the basic concept behind the Place Bar, but did not find it particularly easy to use. Participants also noted some potential problems as well, such as wanting to input locations that were semantically useful to specific individuals, such as “work” or “home”. We believe that a user interface component along the lines of the Place Bar is still needed, but the current implementation still needs some more revision before it becomes practical and useful.

8.1.4 Peer Enforcement of Privacy Tags

Another area for future work is to implement the peer enforcement of privacy tags, as described in Section 5.4.1. This would require digital signing of privacy tags, as well as some kind of public key infrastructure for checking the validity of signatures.

8.2 Long-Term Future Work

In this section, we look at areas for future work in the long term, including incentives for deploying privacy-sensitive applications, evaluation of changes over time in attitudes and behavior with respect to ubicomp and privacy, better design methods for privacy-sensitive applications, further exploration of the tradeoffs between privacy and locality, better user interfaces to understand disclosures after the fact, more tools for facilitating the creation of privacy-sensitive ubicomp applications, exploring the use of third parties for managing personal privacy, and exploring the overall reliability of ubicomp systems with respect to plausible deniability.

8.2.1 Incentives for Deploying Privacy-sensitive Applications

One important dimension that this dissertation does not address is incentives for inducing companies and open-source developers to build and deploy privacy-sensitive applications. Currently, there are strong economic incentives for companies to disregard consumer privacy. For example, Odlyzko has made the argument that organizations have a strong incentive to collect as much information as possible in order to do price discrimination more effectively [112]. McCullagh makes a related argument, noting that freely flowing personal information has in many cases reduced business transaction costs, and that while this has had negative ramifications on privacy, it has also resulted in benefits to consumers such as faster lines of credit, more efficient services, and convenience [105].

Here, we outline two broad strategies for the technical research community to pursue in developing incentives for privacy-sensitive ubicomp. Roughly speaking, these are the “carrot”, that is benefit to the organization deploying the system, and the “stick”, that is punishment for not deploying privacy-sensitive systems. With respect to the former (the “carrot”), one prospect is to demonstrate to the people who are developing and deploying these systems that there is a relatively low cost for a high amount of benefit. This can be in terms of, for example, lower maintenance costs, better scalability, or better software that is simply easier to deploy. The scalability argument seems to be an especially compelling option to pursue, since, as noted in Section 5.2, support for location at the physical / sensor layer has started to move from centralized location-tracking systems towards decentralized location-support ones, primarily for reasons of scale. Continued work by the research community along these lines would give ubicomp a persuasive value proposition for both developers (scalability and maintainability) and end-users (privacy), making it a win-win situation for all stakeholders.

With respect to the latter (the “stick”), one possibility here is to pollute the data, making it harder to trace specific individuals. In a short story, science fiction author Vernor Vinge described how a group called “the Friends of Privacy” polluted the web so badly with false information about individuals that it was difficult to sort fact from fiction [143]. Another possibility, one that is currently being pursued by Intel Research Seattle, is to have a special license on the source code. Similar to the GNU general public license, this license would require people using this code to comply with several privacy principles.

8.2.2 Changes in Attitudes and Behaviors over Time

The notion of information privacy is a relatively modern concept, one that is also constantly being re-formulated as new technologies become widespread and embedded in everyday activities. Some technologies initially perceived as intrusive are now commonplace and even seen as desirable, clearly demonstrating that peoples’ attitudes and behaviors towards a technology can change over time.

For example, in the book *Calling America*, Fischer describes the history of the telephone, noting that at first, many people objected to having phones in their homes because it “permitted intrusion… by solicitors, purveyors of inferior music, eavesdropping operators, and even wire-transmitted germs” [55]. While these were real concerns expressed by people back then, by modern standards, this view would probably be seen as overly paranoid.

Similar concerns were expressed when the Kodak camera, the first easy-to-use camera that could take near instant photos, was introduced in 1888. Journalist David Lindsay writes:

> The appearance of Eastman's cameras was so sudden and so pervasive that the reaction in some quarters was fear. A figure called the "camera fiend" began to appear at beach resorts, prowling the premises until he could catch female bathers unawares. One resort felt the trend so heavily that it posted a notice: "PEOPLE ARE FORBIDDEN TO USE THEIR KODAKS ON THE BEACH." Other locations were no safer. For a time, Kodak cameras were banned from the Washington Monument. The "Hartford Courant" sounded the alarm as well, declaring the "the sedate citizen can't indulge in any hilariousness without the risk of being caught in the act and having his photograph passed around among his Sunday School children." [101]

These anecdotes and informal observations were the insights that led to our working hypothesis that the acceptance of many potentially intrusive technologies follows a curve that we call "the privacy hump" (see Figure 8-1.). Early on in the life cycle of a technology, there are many fears and concerns about how these technologies will be used. Some of these are legitimate concerns, while others are based more on misunderstandings about the technology (for example, the quote above that phones could transmit germs). There are also many questions about the right way of deploying these technologies. Businesses have not worked out how to convey the right value propositions to consumers, and society has not worked out what is and is not acceptable use of these technologies. These fears are often conceptualized under the rubric of "privacy", forming a "privacy hump" that represents a barrier to the acceptance of a potentially intrusive technology.

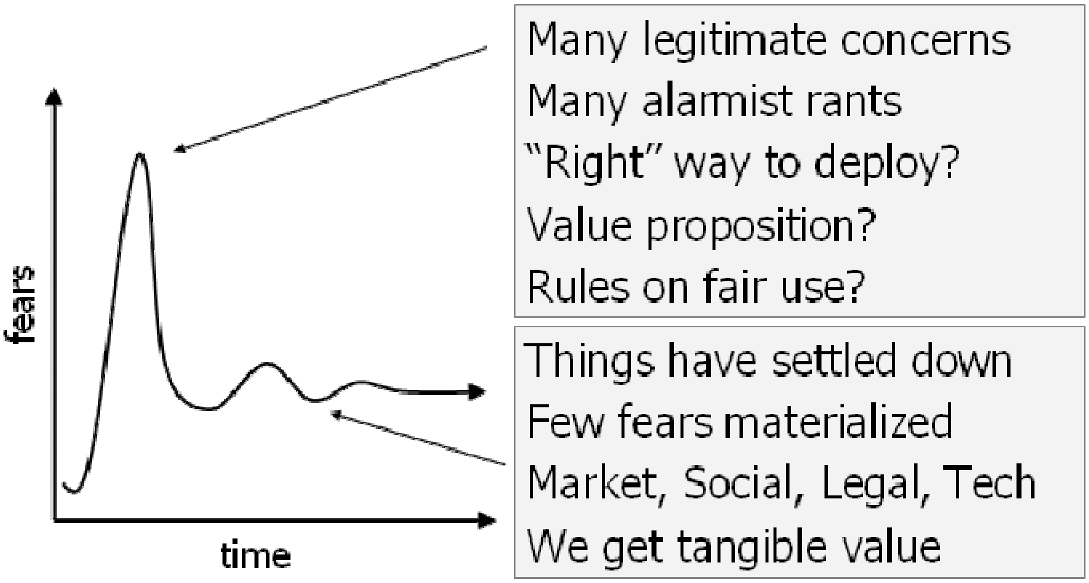


Figure 8-1. One working hypothesis we have developed describing the acceptance of potentially intrusive technologies is the "privacy hump". Early in the life cycle of a technology, there are many fears and concerns about how that technology will be used, often couched in terms of privacy. However, if, over time, privacy violations have not occurred, and if the entire system of market, social, legal, and technical forces have adapted to address legitimate concerns, then a community of users can overcome this privacy hump.

Over time, however, the factors contributing to these fears start to work themselves out. This could be because the fears did not materialize (for example, very few phone companies send inferior music to us), society has adapted itself to the technology (for example, most people understand it is appropriate to take a photo at a wedding but not at a funeral), or laws are passed to punish violators (for example, the do not call list protecting individuals from telemarketers or laws designed to punish peeping toms). In other words, if a community of users overcomes the "privacy hump", it is not because their privacy concerns have disappeared, but because parts of the entire system–the market, social norms, laws, and technology–have adapted to make these concerns understandable and manageable. It should be noted, however, that the privacy hump is not always overcome simply with the passage of time. For example, as we have described before, nurses have rejected the use of locator badges in more than one instance [71, 123].

This hypothesis is still speculation at this point, and it is not immediately obvious to us how to acquire empirical evidence to confirm or refute it. However, if it is somewhat accurate as a predictive model, it suggests many potential directions for future research. For example, what factors contribute to the fears expressed by a community of users? What steps can developers of ubicomp technologies take to flatten the peak of the privacy hump, to accelerate the process of acceptance (assuming that a given technology should be accepted)? How does experience affect individual conceptions of privacy? For example, preliminary results from a study conducted by Pew Internet & American Life suggests that when people first use the Internet, they are less likely to do risky activities such as buying things online or talking with strangers, but are more likely to do so after a year of experience [119]. Understanding the privacy hump from these perspectives would be useful, as it would help us understand how to design and deploy technologies better and increase the likelihood that a technology is accepted.

8.2.3 Design Methods for Privacy-sensitive Applications

Most discussions about privacy usually generate more heat than light, often because people have very different and individualistic notions of privacy. This lack of common grounding makes it difficult to have reasoned debates as to what the potential risks are, and what potential solutions can be applied to address those risks.

We believe that the research community and the design community need to work together in developing better methods for helping practitioners understand this design space and come up with effective solutions. Here, we describe two different directions that we have taken to address this problem, namely privacy risk models and design patterns for ubiquitous computing.

The main idea behind a privacy risk model is that there should be a systematic method 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 have developed an initial privacy risk model specifically for ubiquitous computing [78]. This privacy risk model helps developers understand and prioritize potential privacy risks by posing a series of questions that commonly arise when developing ubicomp systems. These include: who are the users? What is their relationship? How is personal information collected? Is it shared continuously or discretely? What is the granularity of information shared (for example, with location, it could be city or street level)?

We have also developed an initial set of design patterns to help developers create useful, usable, and privacy-sensitive ubicomp systems [36]. *Design patterns* have been proposed in many domains as a format for capturing and sharing design knowledge between practitioners (e.g., [11, 18, 24, 32, 142]). Patterns communicate insights into design problems, capturing the essence of recurring problems and their solutions in a compact form. Patterns describe the problem in depth, the rationale for the solution, how to apply the solution, and some of the trade-offs in applying the solution. The idea here is that, rather than re-inventing an existing solution, a designer should be able to look up a solution that others have developed and understand the tradeoffs involved.

Several of the patterns we have developed deal explicitly with end-user privacy. We have also conducted empirical evaluation of these patterns with sixteen pairs of designers, to understand how patterns affect the design process. One difficulty we encountered, however, is that many of the designers understood that privacy was an important consideration for ubicomp applications, but very few actually used our patterns to come up with solutions to address it. We are currently looking at several reasons as to why this happened, so that we can revise the patterns to make them more effective for privacy.

8.2.4 Further Exploration of Tradeoffs between Privacy and Locality

Confab is structured such that information is processed locally on an end-user's computer as much as possible. As noted previously, there are interesting tradeoffs here between privacy, consistency and freshness of data, computational and storage requirements of the client, as well as overall deployability.

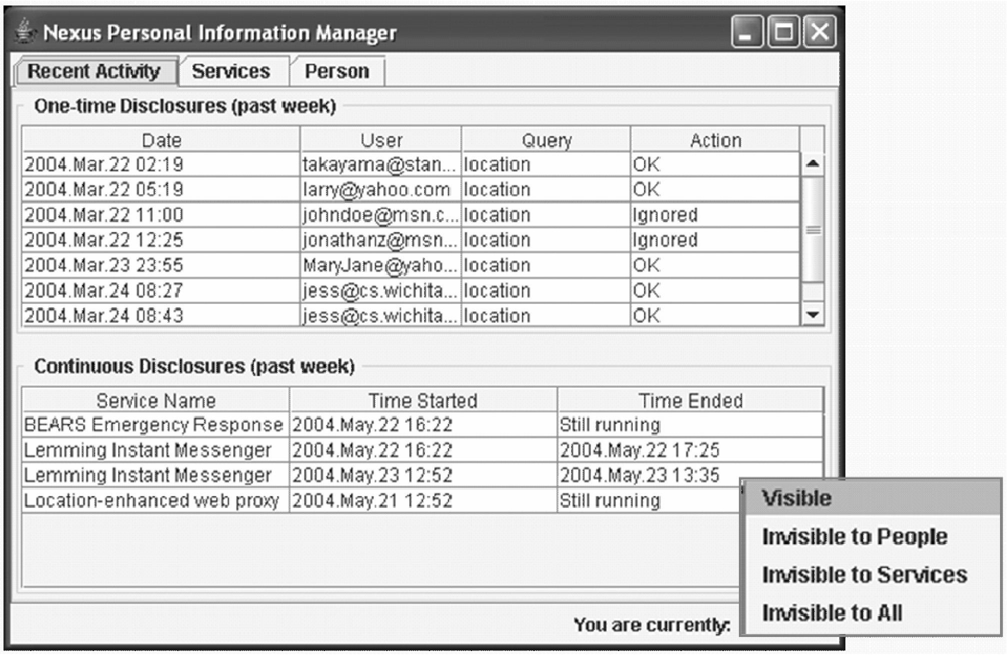
For example, with Confab, we use locality for reasons of privacy. However, this means that sometimes the data on that device must be periodically updated (for example, updating the Place Lab access point database, as well as the places database). This approach also requires smarter clients, in that it is expected that end-user’s clients have reasonable processing power and storage capability, which also cost more in terms of money and power consumption. This approach also shifts the burden of system administration onto the end-user. A final consideration here is the risk of accidental disclosure or deletion of data by end-users, as well as malicious attacks through viruses, Trojan horses, and social engineering.

On the other hand, locality means that systems are somewhat easier to deploy, because there is less infrastructure that needs to be set up, and because there are fewer dependencies on other systems and thus fewer possible chains of failure. A failure in one part of the system will not necessarily bring down the whole system, as is the case with centralized systems.

There is a rich design space to explore here. For example, what approaches are there to ensure better privacy for thin clients? How can network proxies be used to lower the power and storage needs? Can personal data be stored in encrypted formats in the network and be just as effective? How useful would network proxies be in terms of privacy? How often do locally stored databases have to be updated to be effective? Are there other ways locality can be used to improve or accelerate the deployability of ubicomp systems? Is it possible to hybridize centralized and decentralized systems, so that systems can always work independently but can easily federate with other available systems to form more effective and robust systems? As an example, in past work on emergency response, we designed a system that offered useful sensor information to individual firefighters and could also automatically share this information with other nearby firefighters [83].

8.2.5 Better User Interfaces to Understand Disclosures after the Fact

In Chapter 5, we described several user interfaces for helping people manage their privacy. Two of these user interfaces, the access notification and the Place Bar, are meant to help people make decisions about sharing. A third user interface, shown in Figure 5-10 and reproduced below as Figure 8-2, shows who has requested what information, as well as what services are currently active. Currently, this user interface is a simple proof of concept and has not been user tested for usefulness or usability. One could imagine better summaries and better visualizations to show how information is flowing to others.



One-time Disclosures (past week)

Date	User	Query	Action
2004.Mar.22 02:19	takayama@stan...	location	OK
2004.Mar.22 05:19	larry@yahoo.com	location	OK
2004.Mar.22 11:00	johndoe@msn.c...	location	Ignored
2004.Mar.22 12:25	jonathanz@msn...	location	Ignored
2004.Mar.23 23:55	MaryJane@yaho...	location	OK
2004.Mar.24 08:27	jess@cs.wichita...	location	OK
2004.Mar.24 08:43	jess@cs.wichita...	location	OK

Continuous Disclosures (past week)

Service Name	Time Started	Time Ended
BEARS Emergency Response	2004.May.22 16:22	Still running
Lemming Instant Messenger	2004.May.22 16:22	2004.May.22 17:25
Lemming Instant Messenger	2004.May.23 12:52	2004.May.23 13:35
Location-enhanced web proxy	2004.May.21 12:52	Still running

Figure 8-2. This UI shows who has requested what information. It also provides a simple way of going into invisible mode for just people, just services, or to everything.

Generally speaking, this user interface needs to support three high level tasks, namely, who in theory can access one’s personal information, who actually has in the past, and who currently is. This user interface should also support the addition or revision of any rules that end-users may want to place on access. One possibility here is to show previews of how access will change. For example, one could imagine using a person’s actual history, showing the current access privileges and how those access privileges will change given a new policy. This approach would help ground the end-user, letting them see in a concrete way how their daily activities would be perceived by others, rather than abstracting it as a simple rule.

8.2.6 Tools for Facilitating the Creation of Ubicomp Applications

Confab is currently designed with the expectation that application developers will be skilled in the craft of systems programming. While this is a reasonable first step for a research project meant to demonstrate privacy goals rather than programmability goals, it still poses a significant barrier to entry for the population at large. Simplifying Confab for a non-trivial but useful subset of ubicomp applications could have significant impact, in the same way that the simple content authoring model for the World Wide Web has led to its widespread success. This line of research would also make several contributions to the research community, including a stronger demonstration of the feasibility and effectiveness of Confab's data model and program model, a more rigorous evaluation of Confab's privacy model, as well as the practical utility of getting more ubicomp applications out there.

One direction we have already taken along these lines is prototyping tools for ubiquitous computing. We have helped develop a tool called Topiary [100], a rapid prototyping tool for location-enhanced applications. Topiary lets designers quickly create mockups of interaction sequences that make use of location information (for example, "show this page when John enters room 525"), and then test those mockups using a Wizard of Oz approach where a person fakes location information. Topiary provides three advantages over existing approaches for creating location-enhanced applications. First, it lowers barriers to entry, making it easier for interaction designers who are not experts in the underlying technologies to take part in development. Second, it helps speed up iterative design cycles by making it easier to design, prototype, and evaluate ideas. Third, it makes it easier to get user feedback early in the design cycle, when it is still cheap and relatively simple to make major changes.

One could imagine integrating Topiary with Confab, turning Topiary into an authoring tool for creating actual ubicomp applications rather than just a tool for creating and testing mockups. The benefit here is that this new version of Topiary would be able to make creating certain kinds of ubicomp applications as easy as creating HTML web pages. This metaphor could also be extended literally by adding extensions to HTML so that it can make use of implicit sensor input, such as location or activity information. By leveraging a content model that many people are already familiar with, this approach would make it relatively quick and easy to create and deploy simple kinds of ubicomp applications. It also has the advantage of making these applications very easy to deploy, though certain steps would need to be taken to protect privacy in this model, as web servers could easily track where a person is going based on the pages retrieved (for example, file `abcd.html` is retrieved only if the person enters a Starbucks café, so we now know that they are in a cafe). Pre-fetching (i.e., retrieving large quantities of potentially useful data beforehand), chaff (i.e., retrieving random pages to add noise to the data), and proxies that fill in sensitive information (i.e., a trusted edge service that fills in, for example, local points of interest, right before the content goes to the end-user) are three possibilities for overcoming this problem.

Another promising direction for simplifying the creation of ubicomp applications is to create specialized end-user programming tools based on events. Previous work in end-user programming for children [117] strongly suggests that events are a natural way of thinking about phenomena (for example, do something interesting "when Cynthia enters the room" or "when the door is opened"). One could imagine a simple tool that would make it easy to glue existing systems together via events. For example, "when the alarm clock rings, start the coffee maker" or "when the laundry is done, send a text message to me". If it were done in a simple and easy enough manner, this would let people combine existing ubicomp systems in ways that are useful for them. In many respects, this is similar to how calendar programs let people create alarms that bring up reminders when those events occur. The end-user programming proposed here expands this same basic idea to the vaster design space of ubiquitous computing.

8.2.7 Third Parties for Managing Personal Privacy

One very intriguing possibility is the development of third-party companies that can help store and manage one's personal information for them. Earlier, we described how MedicAlert [2] is an example of such an organization. Such companies could be non-profit, reducing the economic incentive to misuse one's data, or could be for-profit, providing an economic incentive and possibly a legal obligation to manage one's data properly.

Some possibilities include:

- a company that tries out various services and assesses them, providing users with a clearinghouse of ratings to make it easier to understand what providers to trust
- a service similar to BEARS (as described in Section 6.1.3) that holds one's actual location information and only discloses it in case of emergencies
- a service that seeds other services with fake data and tracks how that data is used, making it easier to see abuses such as price discrimination or location-based spam
- a service that helps do data mining on your own information, making it easier for end-users to understand what kinds of information a company might discover if they disclose a certain piece of information (for example, Acme corporation knows X and Y about you, but if you give them Z, they will also be able to infer A, B, and C as well).

8.2.8 Overall Reliability of Ubicomp Systems and Plausible Deniability

One philosophical question that this dissertation raises is, how reliable do we want ubicomp systems to be? Confab relies on the fact that there will be some level of ambiguity at the physical / sensor layer to provide a level of plausible deniability.

However, as the underlying systems become more widely deployed and more effective at sensing and fusing, the amount of plausible deniability is reduced. In other words, it is possible that we may not want a perfect ubicomp system, one that provides no place to hide, no room for ambiguity, no possibility of white lies.

In the near-term, this will not be an issue, since there will be many privacy-protecting obstacles with respect to cost of deployment, reliability of sensors, and administrative domains. However, this will almost certainly be an issue in the long-term, one without a clear answer. One possibility is to deliberately design ubiquitous computing systems with certain intrinsic inefficiencies. For example, Lessig has argued that this is one possible approach, making an analogy with how democratic governments are designed to have checks and balances, a deliberate inefficiency meant to protect citizens against the tyranny of government [98]. Although it is clearly speculation at this point, it is possible that ubiquitous computing might evolve along the same lines for precisely the similar reasons.

8.3 Summary

In this chapter, we looked at future work for both the short-term and the long-term. Areas of interest for the short term include continued development and evaluation of ubicomp applications, better integration of access notifications with instant messengers, developing alternative user interfaces to the place bar, and implementation of peer enforcement of privacy tags.

Areas of interest for the long term include incentives for deploying privacy-sensitive applications, evaluation of changes over time in attitudes and behavior with respect to ubicomp and privacy, better design methods for privacy-sensitive applications, further exploration of the tradeoffs between privacy and locality, better user interfaces to understand disclosures after the fact, more tools for facilitating the creation of privacy-sensitive ubicomp applications, exploring the use of third parties for managing personal privacy, and exploring the overall reliability of ubicomp systems with respect to plausible deniability.

9 Conclusions

The key problem that this dissertation addresses is that it is difficult to create privacy-sensitive ubicomp applications. To address this, we presented the design, implementation, and evaluation of Confab, a toolkit that facilitates the construction and deployment of high-quality privacy-sensitive ubiquitous computing applications.

This dissertation makes four major research contributions. The first three of these contributions address important problems in developing privacy-sensitive ubiquitous computing applications. The first problem is that it is hard to *analyze* end-user needs for ubicomp privacy. Towards this end, we presented a *comprehensive set of end-user needs* gathered from a variety of sources. These included scenario-based interviews that we conducted to understand the range of privacy concerns with respect to ubicomp applications, an analysis of freeform comments from a survey on ubicomp privacy preferences, an investigation of postings on a nurse message board describing experiences using locator systems, a synthesis of previously reported experiences with ubicomp systems, and an examination of proposed and existing privacy laws. This set of needs is useful in informing designers of the range of privacy concerns end-users have with ubicomp systems.

The second problem is that it is difficult to *design* effective user interfaces for ubicomp privacy. Towards this end, we described *a set of pitfalls in designing user interfaces for ubicomp privacy, derived from an analysis of over forty different* *applications for common mistakes still being made*. These pitfalls are useful in informing designers of common user interface mistakes and ways of avoiding those mistakes.

The third problem is that it is difficult to *build* privacy-sensitive ubicomp applications. Towards this end, we presented the *design, implementation, and evaluation of the Confab toolkit*. Based on the set of end-user needs and analysis of user interface pitfalls described above, Confab facilitates the construction of privacy-sensitive ubicomp applications by providing an extensible framework for capturing, processing, and presenting personal information. Confab introduces the idea of protection for ubicomp privacy at the physical, infrastructure, and presentation layers. Confab also introduces an alternative architecture for ubicomp applications, where personal information is captured, stored, and processed as much as possible on computers that end-users have control over, along with user interfaces for helping end-users make better decisions about disclosures. This is in contrast to previous architectures for ubicomp which have tended to distribute capture, storage, and processing over the network, making it harder for end-users to control the flow of their personal information.

The fourth contribution of this work is an evaluation of this toolkit through building three novel applications and informal user studies of those applications. These include a location-enhanced instant messenger, a location-enhanced web proxy, and an emergency response application. We also conducted user studies with nine people of the first two of these applications. These user studies provided preliminary evidence that people could understand the user interfaces at a conceptual level, could share personal information at a desired level, that most users assumed that the location information started with them (regardless of whether this was true or not), and were quite interested about using two of the three applications, namely the location-enhanced instant messenger and the location-enhanced web proxy.

References

1. *AllNurses.com*. http://allnurses.com/

2. *MedicAlert*. http://www.medicalert.org

3. *Preliminary Comments of the Electronic Privacy Information Center, in the Public Workshop--Monitoring Software on Your PC: Spyware, Adware, and Other Software*, in *Federal Trade Commission*. 2004: Washington, D.C. http://www.ftc.gov/os/comments/spyware/040419epic.pdf

4. *Wireless Privacy Protection Act of 2003*. 2003. http://www.theorator.com/bills108/hr71.html

5. Abowd, G.D., et al., *Cyberguide: A Mobile Context-Aware Tour Guide.* Baltzer/ACM Wireless Networks, 1997. **3**(5): pp. 421-433.

6. Ackerman, M., *The Intellectual Challenge of CSCW: The Gap Between Social Requirements and Technical Feasibility*, in *HCI in the New Millennium,* J. Carroll, Editor. 2001, Addison-Wesley.

7. Adams, A. *Multimedia Information Changes the Whole Privacy Ball Game*. In *Computers, Freedom, and Privacy*. Toronto, Canada: ACM Press. pp. 25-32. 2000.

8. Addlesee, M., et al., *Implementing a Sentient Computing System.* IEEE Computer, 2001. **34**(8): pp. 50-56.

9. Agrawal, R. and R. Srikant. *Privacy-Preserving Data Mining*. In *ACM SIGMOD Int'l Conf. on Management of Data*. Dallas, Texas. pp. 2000.

10. Agre, P.E. and M. Rotenberg, *Technology and Privacy: The New Landscape*. 1997, Cambridge MA: MIT Press.

11. Alexander, C., *A Pattern Language: Towns, Buildings, Construction*. 1977: Oxford University Press.

12. allnurses.com, *New call lights*. 2001. http://allnurses.com/forums/showthread.php?t=19798&page=1&pp=10

13. allnurses.com, *New Restroom protocol per management.* 2002. http://allnurses.com/t16164.html

14. allnurses.com, *Nurse Tracking Devices: Whats Your Opinion?* 2001. http://allnurses.com/t8012.html

15. AT&T, *AT&T Wireless mMode - Find Friends*. http://www.attwireless.com/mmode/features/findit/FindFriends/

16. Baertlein, L., Calif. Lawmaker Moves to Block Google's Gmail, *Reuters* pp., 2004.

17. Barkhuus, L. and A.K. Dey. *Location-based services for mobile telephony: a study of users' privacy concerns*. In *INTERACT 2003, 9th IFIP TC13 International Conference on Human-Computer Interaction*. pp. To appear. 2003.

18. Bayle, E., et al., Putting it all together: towards a pattern language for interaction design, *SIGCHI Bulletin*, vol. 30(1): pp. 17-23, 1998.

19. BBC News, *Radio tags spark privacy worries*. 2003. http://news.bbc.co.uk/1/hi/technology/3224920.stm

20. Beckwith, R., *Designing for Ubiquity: The Perception of Privacy.* IEEE Pervasive, 2002. **2**(2): pp. 40-46.

21. Bellotti, V. and A. Sellen. *Design for Privacy in Ubiquitous Computing Environments*. In *The Third European Conference on Computer Supported Cooperative Work (ECSCW'93)*. Milan, Italy: Kluwer Academic Publishers. pp. 1993.

22. Bentham, J., *The Panopticon Writings*, in *The Panopticon and Other Prison Writings*, M. Bozovic, Editor. 1995. pp. 29-95. http://cartome.org/panopticon2.htm

23. Beresford, A. and F. Stajano, Location Privacy in Pervasive Computing, *IEEE Pervasive Computing*, vol. 2(1): pp. 46-55, 2003.

24. Borchers, J., *A Pattern Approach to Interaction Design*. 2001: John Wiley and Sons. http://hcipatterns.org

25. boyd, d., *Faceted Id/entity: Managing representation in a digital world*, Unpublished Master's Thesis, MIT, MIT Media Lab, Cambridge, MA, 2002.

26. Brin, D., *The Transparent Society*. 1998, Reading, MA: Perseus Books.

27. Brown, P.J. and G.J.F. Jones, *Context-aware Retrieval: Exploring a New Environment for Information Retrieval and Information Filtering.* Personal and Ubiquitous Computing, 2001. **5**(4): pp. 253-263.

28. Burrell, J., T. Brooke, and R. Beckwith. *Extending Ubiquitious Computing to Vineyards*. In *Extended Abstracts on Human Factors in Computing Systems (CHI2003)*. pp. 822-823. 2003.

29. Burrell, J., et al. *Context-Aware Computing: A Test Case*. In *Ubicomp 2002*. Göteborg, Sweden. pp. 1-15. 2002.

30. Cadiz, J. and A. Gupta, *Privacy Interfaces for Collaboration*. 2001, Microsoft Research: Redmond, WA. http://www.research.microsoft.com/research/coet/Privacy/TRs/01-82.pdf

31. California Nurses Association, *Eden RNs Protest Electronic Tracking Devices: Mass Turn-in of Nurse Locator Buttons*. 2002. http://www.calnurse.org/cna/press/90402a.html

32. Casaday, G. *Notes on a Pattern Language for Interactive Usability*. In *Human Factors in Computing Systems: CHI 1997*. Atlanta, GA. pp. 289-290. 1997.

33. Castro, P. and R. Muntz, *Managing Context for Smart Spaces.* IEEE Personal Communications, 2000. **5**(5). http://godfather.CS.UCLA.EDU/publications/pdf/ieeePCAug.pdf

34. Chaum, D., Untraceable Electronic Mail, Return Addresses, and Digital Pseudonym, *Communications of the ACM*, vol. 24(2): pp. 84-88, 1981.

35. Chen, G. and D. Kotz. *Context Aggregation and Dissemination in Ubiquitous Computing Systems*. In *Fourth IEEE Workshop on Mobile Computing Systems and Applications*. pp. 105-114. 2002.

36. Chung, E.S., et al. *Development and Evaluation of Emerging Design Patterns for Ubiquitous Computing*. In *Designing Interactive Systems (DIS2004)*. Boston, MA. pp. 233-242. 2004.

37. Computing Research Association, *CRA Conference on "Grand Research Challenges in Information Security & Assurance"*. 2003. http://www.cra.org/Activities/grand.challenges/security/

38. Computing Research Association, *CRA Conference on "Grand Research Challenges" in Computer Science and Engineering*. 2002. http://www.cra.org/Activities/grand.challenges/

39. Coy, P., Big Brother, Pinned to your chest, *Business Week*, vol. pp., 1992.

40. Cranor, L., et al., *The Platform for Privacy Preferences 1.0 (p3p1.0) specification.* 2000, W3C. http://www.w3.org/TR/P3P/

41. Crowley, J.L., et al. *Perceptual Components for Context Aware Computing*. In *Ubicomp 2002*. Göteborg, Sweden. pp. 117-134. 2002.

42. Cuellar, J., et al., *Geopriv requirements (Internet Draft)*. 2003, IETF. http://www.ietf.org/internet-drafts/draft-ietf-geopriv-reqs-04.txt

43. Davies, N., et al. *Limbo: A tuple space based platform for adaptive mobile applications*. In *The International Conference on Open Distributed processing / Distributed Platforms (ICODP/ICDP '97)*. pp. 291-302. 1997.

44. Dey, A., *Providing Architectural Support for Building Context-Aware Applications*, Unpublished, Georgia Institute of Technology, College of Computing, Atlanta, GA, 2000. http://www.cc.gatech.edu/fce/ctk/pubs/dey-thesis.pdf

45. Dey, A.K., D. Salber, and G.D. Abowd, *A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications.* Human-Computer Interaction (HCI), 2001. **16**(2-3): pp. 97-166.

46. Doheny-Farina, S., The Last Link: Default = Offline, Or Why Ubicomp Scares Me, *Computer-mediated Communication*, vol. 1(6): pp. 18-20, 1994.

47. Edwards, J., *Location Privacy Protection Act of 2001*. 2001. http://www.techlawjournal.com/cong107/privacy/location/s1164is.asp

48. Edwards, W.K., et al. *Challenge: Recombinant Computing and the Speakeasy Approach*. In *Eighth ACM International Conference on Mobile Computing and Networking (MobiCom 2002)*. pp. 279-286. 2002.

49. Espinoza, F., et al. *GeoNotes: Social and Navigational Aspects of Location-Based Information Systems*. In *Ubicomp 2001*. Atlanta, GA. pp. 2-17. 2001.

50. Etzioni, A., *The Limits of Privacy*. 1999, New York: Basic Books.

51. European Union, *Directive 95/46/EC*. 1995. http://europa.eu.int/ISPO/legal/en/dataprot/directiv/directiv.html

52. Falk, J., et al. *Pirates: Proximity-Triggered Interaction in a Multi-Player Game*. In *Human Factors in Computing Systems: CHI 2001 (Extended Abstracts)*. pp. 119-120. 2001.

53. Federal Communications Commission, *Enhanced 911*. http://www.fcc.gov/911/enhanced/

54. Fielding, R., *Architectural Styles and the Design of Network-based Software Architectures*, Unpublished PhD dissertation, University of California at Irvine, Incormation and Computer Science, Irvine, CA, 2000.

55. Fischer, C., *America Calling*. 1994: University of California Press.

56. Foucault, M., *Discipline and Punish*. 1977, New York, NY, USA: Vintage Books.

57. Friedman, B., D.C. Howe, and E. Felten. *Informed Consent in the Mozilla Browser: Implementing Value-Sensitive Design*. In *The Thirty-Fifth Annual Hawai'i International Conference on System Sciences*: IEEE Computer Society. pp. CD-ROM of full-paper, OSPE101. 2002.

58. Gardner, D., *Urban Wi-Fi Gridlock Predicted To Arrive in 2004*. 2003. http://www.techweb.com/wire/26802643

59. Garfinkel, S., *Database Nation: The Death of Privacy in the 21st Century*. 2001: O'Reilly & Associates.

60. Geocaching. http://www.geocaching.com/

61. Goffman, E., *The Presentation of Self in Everyday Life*. 1959, New York: Anchor, Doubleday.

62. Goldstein, H., Mike Villas's World, *IEEE Spectrum*, vol. 41(7): pp., 2004.

63. Good, N. and A. Krekelberg, *Usability and Privacy: A study of Kazaa P2P file-sharing.* CHI Letters, 2003. **5**(1): pp. 137-144.

64. Google, *About Gmail*. http://gmail.google.com/gmail/help/about.html

65. Grimm, R., et al., *System support for pervasive applications.* ACM Transactions on Computer Systems, 2004. **22**(4): pp. 421-486.

66. Griswold, W.G., et al., *ActiveCampus - Experiments in Community-Oriented Ubiquitous Computing*. 2003, Computer Science and Engineering, UC San Diego.

67. Grudin, J., *Desituating Action: Digital Representation of Context.* Human-Computer Interaction (HCI) Journal, 2001. **16**(2-4).

68. Grudin, J., Groupware and Social Dynamics: Eight Challenges for Developers, *Communications of the ACM*, vol. 37(1): pp. 92-105, 1994.

69. 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

70. Gruteser, M. and D. Grunwald. *Anonymous Usage of Location-Based Services Through Spatial and Temporal Cloaking*. In *The First International Conference on Mobile Systems, Applications, and Services (MobiSys 2002)*. pp. 2002.

71. Harper, R.H. *Why People Do and Don't Wear Active Badges: A Case Study*. In *Computer Supported Cooperative Work (CSCW96)*. pp. 297-318. 1996.

72. Harper, R.H.R., M.G. Lamming, and W.N. Newman, *Locating Systems at Work: Implications for the Development of Active Badge Applications.* Interacting with Computers, 1992. **4**(3): pp. 343-363.

73. Harter, A. and A. Hopper, *A Distributed Location System for the Active Office.* IEEE Network, 1994. **8**(1). ftp://ftp.uk.research.att.com/pub/docs/att/tr.94.1.pdf

74. Heer, J., et al. *liquid: Context-Aware Distributed Queries*. In *Fifth International Conference on Ubiquitous Computing: Ubicomp 2003*. Seattle, WA: Springer-Verlag. pp. 140-148. 2003.

75. Hindus, D., et al., *Casablanca: Designing Social Communication Devices for the Home.* CHI Letters, 2001. **3**(1): pp. 325-332.

76. Hong, J.I., et al. *Privacy and Security in the Location-enhanced World Wide Web*. In *Fifth International Conference on Ubiquitous Computing: Ubicomp*

2003 (Workshop on Ubicomp Communities: Privacy as Boundary Negotiation). Seattle, WA. pp. 2003.

77. Hong, J.I. and J.A. Landay. *An Architecture for Privacy-Sensitive Ubiquitous Computing*. In *The Second International Conference on Mobile Systems, Applications, and Services*. Boston, MA. pp. 177-189. 2004.

78. Hong, J.I., J. Ng, and J.A. Landay. *Privacy Risk Models for Designing Privacy-Sensitive Ubiquitous Computing Systems*. In *Designing Interactive Systems (DIS2004)*. Boston, MA. pp. 91-100. 2004.

79. Hull, R., et al. *Enabling Context-Aware and Privacy-Conscious User Data Sharing*. In *Mobile Data Management 2004*. pp. 187-198. 2004.

80. IBM Corporation, *Enterprise Privacy Authorization Language (EPAL 1.1)*. 2003. http://www.zurich.ibm.com/security/enterprise-privacy/epal/Specification/

81. Jancke, G., et al., *Linking Public Spaces: Technical and Social Issues.* CHI Letters (Human Factors in Computing Systems: CHI 2001), 2001. **3**(1): pp. 530-537.

82. Jendricke, U. and D. Gerd tom Markotten, *Usability Meets Security: The Identity-Manager as Your Personal Security Assistant for the Internet*, in *16th Annual Computer Security Applications Conference (ACSAC 00)*. 2000: New Orleans, LA, USA.

83. Jiang, X., et al. *Siren: Context-aware Computing for Firefighting*. In *The Second International Conference on Pervasive Computing (Pervasive 2004)*. Vienna, Austria. pp. 87-105. 2004.

84. Jiang, X., J.I. Hong, and J.A. Landay. *Approximate Information Flows: Socially-based Modeling of Privacy in Ubiquitous Computing*. In *Ubicomp 2002*. Göteborg, Sweden. pp. 176-193. 2002.

85. Johanson, B., A. Fox, and T. Winograd, *The Interactive Workspaces Project: Experiences with Ubiquitous Computing Rooms.* IEEE Pervasive Computing, 2002. **1**(2): pp. 67-74.

86. Kaasinen, E., *User Needs for Location-aware Mobile Services.* Personal and Ubiquitous Computing, 2003. **7**(1): pp. 70-79.

87. Kindberg, T. and J. Barton, *A Web-based Nomadic Computing System.* Computer Networks, 2001. **35**(4): pp. 443-456.

88. Klein, J. and L. Vox, *Brave New GPS World*. 2003. http://www.larta.org/lavox/articlelinks/2003/031103_gtx.asp

89. Korba, L. and S. Kenny. *Towards Meeting the Privacy Challenge: Adapting DRM*. In *2002 ACM Workshop on Digital Rights Management*. Washington DC, USA. pp. 2002. http://crypto.stanford.edu/DRM2002/KorbaKennyDRM20021.pdf

90. Kumagai, J. and S. Cherry, Sensors & Sensibility, *IEEE Spectrum*, vol. 41(7): pp., 2004.

91. Lamming, M. and M. Flynn. *Forget-me-not: Intimate computing in support of human memory*. In *FRIEND 21: International Symposium on Next Generation Human Interfaces*. Meguro Gajoen, Japan. pp. 125-128. 1994. http://www.xrce.xerox.com/publis/cam-trs/pdf/1994/epc-1994-103.pdf

92. Langheinrich, M. *A Privacy Awareness System for Ubiquitous Computing Environments*. In *Ubicomp 2002*. Goteberg, Sweden. pp. 237-245. 2002.

93. Langheinrich, M. *Privacy by Design - Principles of Privacy-Aware Ubiquitous Systems*. In *Ubicomp 2001*. Atlanta, GA. pp. 273-291. 2001.

94. Lederer, S., *Designing Disclosure: Interactive Personal Privacy at the Dawn of Ubiquitous Computing*, Unpublished Master of Science, University of California, Berkeley, Computer Science Division, Berkeley, CA, 2003. http://www.cs.berkeley.edu/projects/io/publications/privacy-lederer-msreport-1.01-no-appendicies.pdf

95. Lederer, S., et al., *Personal Privacy through Understanding and Action: Five Pitfalls for Designers*. Personal and Ubiquitous Computing, 2004. **8**(6): pp. 440-454.

96. 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*. Fort Lauderdale, FL. pp. 724-725. 2003.

97. Lemos, R., *Rental-car firm exceeding the privacy limit?* 2001, News.com. http://news.com.com/2100-1040-268747.html

98. Lessig, L. *The Architecture of Privacy*. In *Taiwan NET'98*. Taipei, Taiwan. pp. 1998.

99. Lessig, L., *Code and Other Laws of Cyberspace*. 1999, New York NY: Basic Books.

100. Li, Y., J.I. Hong, and J.A. Landay, *Topiary: A Tool for Prototyping Location-Enhanced Applications.* CHI Letters, 2004. **6**(2): pp. 217-226.

101. Lindsay, D., *The Kodak Camera Starts a Craze*. 2004. http://www.pbs.org/wgbh/amex/eastman/peoplevents/pande13.html

102. LinkSys, *Wireless Technology Comparison Chart.* http://www.linksys.com/edu/wirelessstandards.asp

103. Mackay, W.E. *Triggers and barriers to customizing software*. In *ACM CHI '91 Human Factors in Computing Systems*. New Orleans, LA. pp. 1991.

104. Mayor, M., *New Wireless Device Could Rescue Firefighters*. 2001. http://www.wirelessnewsfactor.com/perl/story/9134.html

105. McCullagh, D., *Database Nation: The upside of "zero privacy"*. 2004. http://reason.com/0406/fe.dm.database.shtml

106. Millett, L.I., B. Friedman, and E. Felten, *Cookies and Web Browser Design: Toward Realizing Informed Consent Online.* CHI Letters, 2001. **3**(1): pp. 46-52.

107. Nagel, K., et al. *The Family Intercom: Developing a Context-Aware Audio Communication System*. In *Ubicomp 2001*. Atlanta, GA. pp. 176-183. 2001.

108. Nardi, B., S. Whittaker, and E. Bradner. *Interaction and Outeraction: Instant Messaging in Action*. In *ACM Conference on Computer Supported Cooperative Work (CSCW2000)*. pp. 79-88. 2000.

109. Necula, G.C. and P. Lee. *Safe Kernel Extensions Without Run-Time Checking*. In *2nd Symposium on Operating Systems Design and Implementation (OSDI '96)*. pp. 1996.

110. Nguyen, D.H. and E.D. Mynatt. *Privacy Mirrors: Making Ubicomp Visible*. In *Human Factors in Computing Systems: CHI 2001 (Workshop on Building the User Experience in Ubiquitous Computing)*. Seattle, WA: ACM Press. pp. 2001.

111. Norman, D.A., *The Design of Everyday Things*. 2002, New York, NY: Basic Books.

112. Odlyzko, A. *Privacy, Economics, and Price Discrimination on the Internet*. In *ICEC2003: Fifth International Conference on Electronic Commerce*: ACM Press. pp. 355-366. 2003. www.dtc.umn.edu/~odlyzko

113. Olsen, D.R., et al., *Cross-modal Interaction using XWeb.* CHI Letters, The 13th Annual ACM Symposium on User Interface Software and Technology: UIST 2000, 2000. **2**(2): pp. 191-200.

114. OnStar. http://www.onstar.com/

115. Palen, L., *Social, Individual and Technological Issues for Groupware Calendar Systems.* CHI Letters: Human Factors in Computing Systems, CHI 99, 1999. **2**(1): pp. 17-24.

116. Palen, L. and P. Dourish, *Unpacking "Privacy" for a Networked World.* CHI Letters, 2003. **5**(1): pp. 129-136. http://guir.berkeley.edu/projects/denim/denim-chi-2000.pdf

117. Pane, J.F., *A Programming System for Children that is Designed for Usability*, Unpublished Ph.D. Thesis, Carnegie Mellon University, Computer Science Department, Pittsburgh, PA, 2002.

118. Pascoe, J. *The Stick-e Note Architecture: Extending the Interface Beyond the User*. In *International Conference on Intelligent User Interfaces*. pp. 261-264. 1997. http://www.cs.ukc.ac.uk/pubs/1997/337/

119. Pew Internet & American Life, *Testimony of Lee Rainie: Director, Pew Internet & American Life Project*. 2001. http://www.pewinternet.org/reports/toc.asp?Report=34

120. Pew Internet & American Life, *Trust and Privacy Online: Why Americans Want to Rewrite the Rules*. 2000. http://www.pewinternet.org/reports/toc.asp?Report=19

121. Povey, D. *Optimistic Security: A New Access Control Paradigm*. In *1999 New Security Paradigms Workshop*. pp. 1999. http://security.dstc.edu.au/staff/povey/papers/optimistic.pdf

122. Priyantha, N.B., A. Chakraborty, and H. Balakrishnan. *The Cricket Location-Support System*. In *MobiCom 2000: The Sixth Annual International Conference on Mobile Computing and Networking*. Boston, Massachusetts: ACM Press. pp. 32-43. 2000.

123. Reang, P., Dozens of nurses in Castro Valley balk at wearing locators, *The Mercury News* pp., 2002. http://www.mercurynews.com/mld/mercurynews/news/local/4015298.htm?1c

124. Rheingold, H., PARC is Back! *Wired*, vol. 2(2): pp., 1994.

125. Rhodes, B. and T. Starner. *The Remembrance Agent: A Continuously Running Automated Information Retrieval System*. In *The First International Conference on The Practical Application of Intelligent Agents and Multi Agent Technology (PAAM '96)*. London, UK. pp. 487-495. 1996. http://rhodes.www.media.mit.edu/people/rhodes/research/Papers/remembrance.html

126. Román, M., et al., *Gaia: A Middleware Infrastructure to Enable Active Spaces*. IEEE Pervasive Computing, 2002. **1**(4): pp. 74-83.

127. Salvador, T. and K. Anderson. *Practical Considerations of Context for Context Based Systems: An Example from an Ethnographic Case Study of a Man Diagnosed with Early Onset Alzheimer's Disease*. In *Ubicomp 2003*. pp. 243-255. 2003.

128. Satyanarayanan, M., Pervasive Computing: Vision and Challenges, *IEEE Personal Communications*, vol. pp. 10-17, 2001.

129. Schilit, B.N., *A Context-Aware System Architecture for Mobile Distributed Computing*, Unpublished PhD, Columbia University, Computer Science Department, 1995. http://seattleweb.intel-research.net/people/schilit/schilit-thesis.pdf

130. Schilit, B.N., N.I. Adams, and R. Want. *Context-Aware Computing Applications*. In *Workshop on Mobile Computing Systems and Applications*. Santa Cruz, CA: IEEE Computer Society. pp. 1994. http://www.fxpal.xerox.com/people/schilit/wmc-94-schilit.pdf

131. Schilit, B.N., et al. *Challenge: Ubiquitous Location-Aware Computing*. In *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.

132. Schilit, B.N., J.I. Hong, and M. Gruteser, Wireless Location Privacy Protection, *Computer*, vol. 36(12): pp. 135-137, 2003.

133. Siewiorek, D., et al. *SenSay: A Context-Aware Mobile Phone*. In *7th IEEE International Symposium on Wearable Computers (ISWC 2003)*. White Plains, NY. pp. 2003.

134. Sloane, L., Orwellian Dream Come True: A Badge That Pinpoints You, *New York Times* pp. 14, 1992.

135. Smith, S.W. and E.H. Spafford, *Grand Challenges in Information Security: Process and Output*. 2003. http://www.computer.org/security/v2n1/j1sec.htm

136. Snoeren, M.A.C., *A Session-Based Architecture for Internet Mobility*, Unpublished PhD Thesis, MIT, Laboratory for Computer Science, Cambridge, 2002.

137. Spreitzer, M. and M. Theimer. *Providing location information in a ubiquitous computing environment*. In *Fourteenth ACM Symposium on Operating System Principles*. Asheville, NC: ACM Press. pp. 270-283. 1993.

138. Sweeney, L., *k-anonymity: a model for protecting privacy.* International Journal on Uncertainty, Fuzziness and Knowledge-based Systems, 2002. **10**(5): pp. 557-570.

139. 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

140. Taylor, H., *Most People Are "Privacy Pragmatists" Who, While Concerned about Privacy, Will Sometimes Trade It Off for Other Benefits*. 2003. http://www.harrisinteractive.com/harris_poll/index.asp?PID=365

141. The Associated Press, Tennessee May Use GPS on Parolees, *Washington Post* pp., 2004. http://www.washingtonpost.com/wp-dyn/articles/A46556-2004Jul13.html

142. van Duyne, D.K., J.A. Landay, and J.I. Hong, *The Design of Sites: Patterns, Principles, and Processes for Crafting a Customer-Centered Web Experience*. 2002, Reading, MA: Addison-Wesley. http://designofsites.com

143. Vinge, V., Synthetic Serendipity, *IEEE Spectrum*, vol. 41(7): pp., 2004.

144. Want, R., et al., *The Active Badge Location System.* ACM Transactions on Information Systems, 1992. **10**(1): pp. 91-102.

145. Want, R., et al., *Overview of the PARCTAB Ubiquitous Computing Experiment.* Mobile Computing, 1995. **2**(6): pp. 28-43.

146. Ward, A., A. Jones, and A. Hopper, *A New Location Technique for the Active Office.* IEEE Personnel Communications, 1997. **4**(5): pp. 42-47.

147. Weiser, M., *The Computer for the 21st Century.* Scientific American, 1991. **265**(3): pp. 94-104.

148. 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): pp. 693-696.

149. Westin, A.F., *Privacy and Freedom*. 1967, New York NY: Atheneum.

150. Whalen, J., You're Not Paranoid: They Really Are Watching You, *Wired Magazine*, vol. 3(3): pp. 95-85, 1995.

151. Whitten, A. and J.D. Tygar. *Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0*. In *8th USENIX Security Symposium*. pp. 1999.

152. Woodruff, A. and P.M. Aoki. *How Push-to-Talk Makes Talk Less Pushy*. In *ACM SIGGROUP Conference on Supporting Group Work (GROUP '03).* Sanibel Island, FL: ACM Press. pp. 170-179. 2003.

Appendix A – Forms used for Scenario-based Interviews

Interview on Location-Based Computing   Interviewee ID:___

My name is Jason Hong, and I am a researcher in the Computer Science Division at the University of California at Berkeley. I would like you to participate in our research, which involves an interview on attitudes toward location-based computing. This interview should take about 45-60 minutes and poses no risks to you other than those normally encountered in daily life.

All of the information that we obtain from your session will be kept confidential. The information obtained from your session will be tagged with a code number. The correspondence between your name and number will be treated with the same care as our own confidential information. We will not use your name or identifying information in any reports of our research.

Your participation in this research is voluntary. You are free to refuse to participate.

If you have any questions about the research, you may call me at 510-643-7354, or send electronic mail to jasonh@cs.berkeley.edu. You may keep the other copy of this form for future reference.

By signing this form you agree to the following statements:
I agree to participate in an interview on location-based computing. I know that the researchers are studying attitudes toward location-based computing. I realize that I will be asked my opinion on several different location-based applications.

I understand that any information obtained during this study will be kept confidential.

I give Jason Hong and his associates permission to present the results of this work in written or oral form, without further permission from me.

Date & Signature

____

Email address for sending gift certificate

____

Age Range
□ 16-20
□ 21-25
□ 26-30
□ 31-35
□ 36-40
□ 41-45
□ 46-50
□ 51+

Gender
□ M □ F

Experience with computers
□ Less than 1 year
□ 1-2 years
□ 3-4 years
□ 5+ years

How would you rate your computer knowledge?
□ Novice
□ Intermediate
□ Expert

Do you own a mobile phone?
□ Y □ N

Do you use instant messenger (for example, Yahoo! or AIM or MSN or Jabber)?
□ Y □ N

Have you used a navigation device before (for example, OnStar Navigation in cars)?
□ Y □ N

Profession (For example, *accountant* or *firefighter*. If student, please indicate major field of study)

____

1. Find a Friend

What Find a Friend offers:

- You can locate nearby friends and find convenient coffee shops for impromptu meetings
- You can add or remove friends (and only friends can request your location)
- Friends can always ask for your current location, and you are always notified whenever any of your friends does so
- You can temporarily go into invisible mode so that nobody sees your location

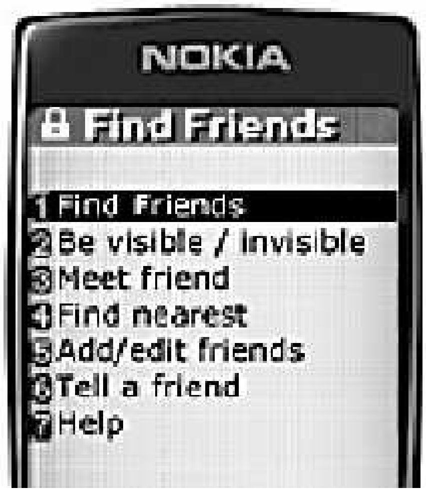


Figure 1. AT&T’s Find Friends interface.

Questions:

1. What people would you be interested in finding?

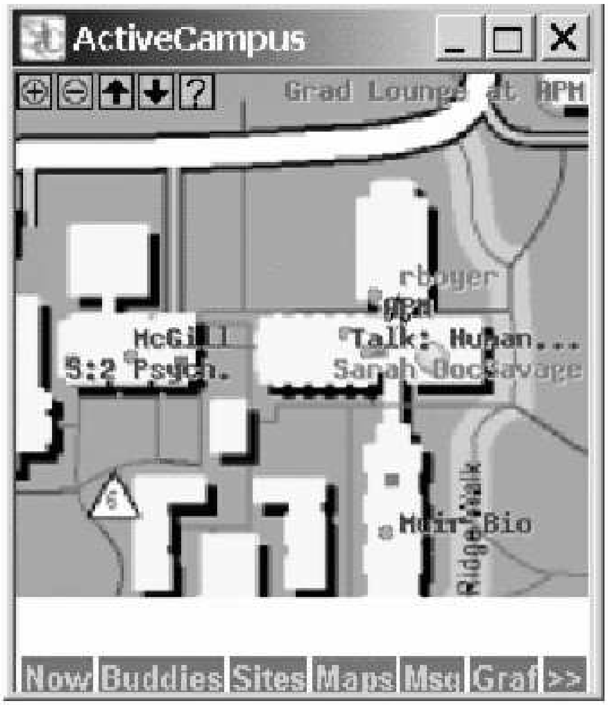
2. Who would be interested in finding you? Also, would you be willing to share your location with friends? Family? Professors and TAs? Roommates? What about someone you just met, for example, a first date? Partners for a class project? Co-workers? Bosses?

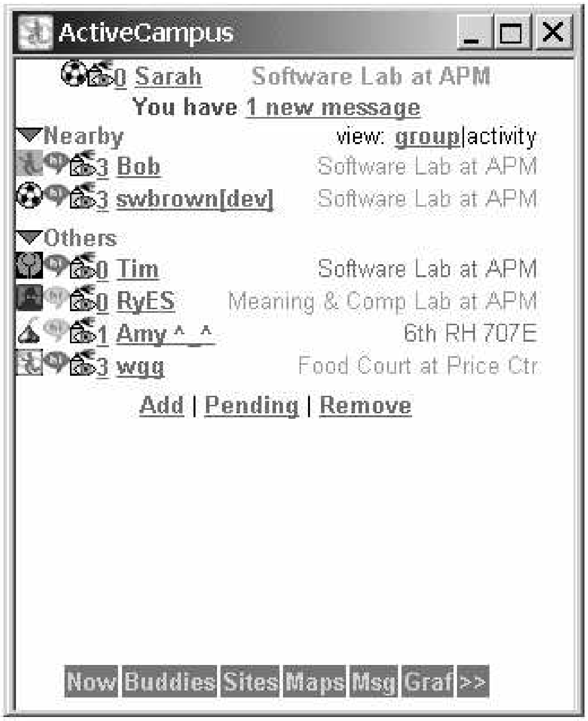
3. What level of location information would you willing to share? City? Street? Building? (For example, “at the corner of Euclid and Hearst”)

4. Would it be better if rather than revealing location, it revealed the general place you were at? For example, “work” or “home” or “café” or “school”?

5. Have you ever used invisible mode for instant messenger? When did you use it? Would those same situations apply here?

2. Active Campus





0 Sarah Software Lab at APM

You have 1 new message

Nearby view: group|activity

3 Bob Software Lab at APM

3 swbrown[dev] Software Lab at APM

Others

0 Tim Software Lab at APM

0 RyES Meaning & Comp Lab at APM

1 Amy ^_^ 6th RH 707E

3 wgg Food Court at Price Ctr

Add | Pending | Remove

Figure 2. The Map (left) shows a map of the user's vicinity, with buddies, sites, and activities overlaid as links at their location. Buddies (right) shows colleagues and their locations, organized by their proximity. Icons to the left of a buddy's name show the buddy on the map.

What Active Campus offers:

- You can see where your friends are on a map, and which friends are nearby
- The map is updated in real-time
- Location information is roughly at the room level within a building
- Your friends can always see your location

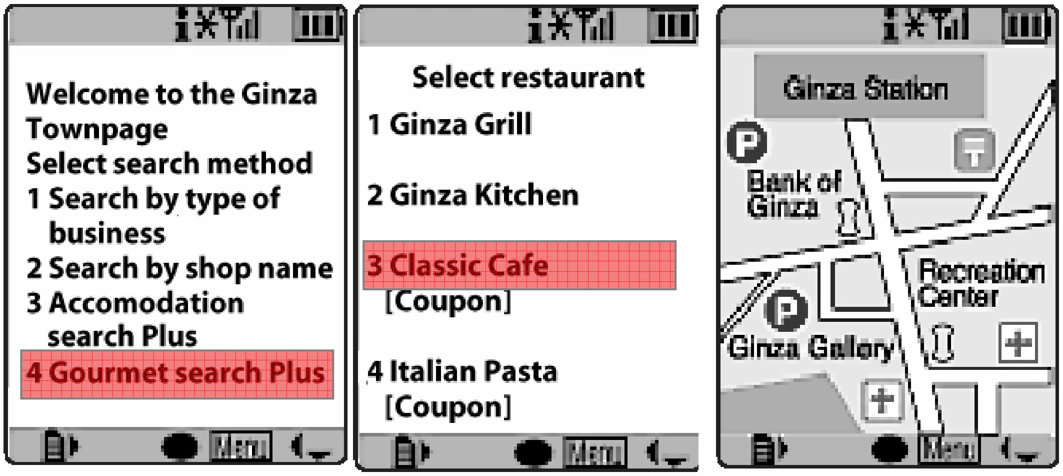
Questions:

1. Would you be willing to share your location with friends? Family? Professors and TAs? Roommates? A first date? Co-workers? Bosses?

2. What if it also revealed your general activity, for example "on the computer" or "playing tennis" or "out with friends"? Would you want this status available to all your buddies or just a subset of buddies?

3. If you use instant messenger, would it be useful to reveal your location to others on your buddy list? What if it also shared activity, such as "in a meeting" or "conducting experiment"?

3. Location-based Searches / Never Get Lost



Welcome to the Ginza Townpage

Select search method

Figure 3. Location-based business searches, eatery guides, and maps.

What it offers:

- Cell-phone networks can locate you and provide searches in your local area, for example, “find me the nearest Mexican restaurants”
- Cell-phone networks can locate you and give personalized directions to places. For example, if you were in London, “how do I get to Big Ben from where I am right now”.

Questions:

1. What kinds of things would you want to look for?
2. How often do you get lost (and need a map or directions)? How do you manage things today if you get lost?
3. If these kinds of searches could use information like your name, home address, and general shopping preferences to give you better results, would you be willing to share this information?
4. Would you be willing to share your name and general shopping preferences to get targeted advertisements from nearby stores? For example, “10% off lunch today, Greek restaurant down the block” or “New CD just came in”

4. Mobile Commerce

What it offers:

- A retail chain pushes information to shoppers depending on their location in a store. For example, if you are near the men’s section, you might get advertisements or coupons for men’s jeans.

- A retail chain also provides online versions of their physical stores, allowing you to search through and navigate what products are in that store. For example, “what kinds of size 4 dresses do they have in this store?” or “tell me where the books by J.D. Salinger are”

Figure 4. A scene from the movie Minority Report, where the protagonist is shopping.

Questions:

1. How useful would getting pushed advertisements be for you?

2. What if the store could link your past purchases to target specific advertisements to you? For example, “we have some new jeans you might like” or “people who bought this shirt also liked these socks”

3. How useful would a physical search engine for a store be for you?

4. What if the store could tailor search results to you? That is, if the store knew your shopping preferences and past purchases, it could order and group things better?

5. Emergency Response Support

What it offers:

- Buildings would know where people were within a building for emergency response purposes, such as fires or earthquakes. Buildings would only know that a person is there rather than who that person was, and the information would be secured so that only a few people could access it.
- To prevent kidnappings, authorities could turn on tracking for your cell phone and then locate where you last were, and possibly where you currently are.
- To improve services, cell phones would automatically transmit your location when making emergency 911 calls

Figure 5. An example emergency response.

Questions:

1. How willing would you be to use a building emergency response service?

2. How willing would you be to use cell phone tracking?

3. How willing would you be to use Emergency 911?

4. How much information are you willing to disclose *before* an emergency happens? Disclosing information beforehand can help in case parts of the system goes down, or your cell phone is damaged for example.

Appendix B – Transcripts from Interviews

This appendix contains partial transcripts from the interviews described in Chapter 2 (Section 2.2.4). The questions and screenshots used are presented in Appendix A. The participants are shown below in a copy of Table 2-3.

Due to time constraints and limited budget, we do not present the full transcripts, but rather field notes and selected key quotes that we feel are representative of that participant's attitudes and tone.

ID	Age	Gender	Computer Skill	Cell	IM	GPS	Profession
1	26-30	M	Expert	Y	Y	N	College Student (CS)
2	21-25	F	Intermediate	Y	Y	N	College Student (Bio)
3	16-20	F	Intermediate	Y	Y	N	College Student (Psych)
4	21-25	F	Intermediate	Y	Y	Y	College Student (Bio)
5	21-25	F	Intermediate	Y	Y	Y	College Student (Comp Lit/Playwriting)
6	21-25	M	Intermediate	N	Y	N	College Student (EECS)
7	21-25	M	Expert	N	Y	N	College Student (CS)
8	51+	M	Expert	N	Y	N	Engineer, Software
9	21-25	F	Intermediate	Y	Y	N	College Student (EECS)
10	21-25	F	Intermediate	Y	Y	Y	Researcher
11	26-30	F	Intermediate	Y	Y	N	Graphic Designer
12	51+	F	Novice	N	N	N	Registered Nurse
13	51+	M	Intermediate	Y	N	Y	Lawyer
14	51+	M	Intermediate	N	N	N	Scientist
15	51+	M	Intermediate	Y	N	N	CEO
16	46-50	F	Intermediate	Y	N	N	Accountant
17	21-25	F	Intermediate	Y	Y	Y	College Student (Math/Economics)
18	16-20	F	Intermediate	N	Y	N	High School Student
19	51+	M	Expert	Y	Y	Y	Systems Engineer
20	21-25	M	Expert	Y	Y	N	Free Lance Web Designer

Demographics of interviewees. Ages were grouped into 5-year ranges, for example 21-25 and 26-30. Column "Computer Skill" was a self-reported indication of whether the interviewee considered themselves a novice, intermediate, or expert with computers. Column "Cell" indicates whether they own a cell phone or not. Column "IM" indicates whether they have used IM before. Column "GPS" indicates whether they have used any electronic navigation device before. All interviewees resided in the San Francisco Bay Area.

ID #1	Age:	26-30
	Gender:	Female
	Computer Skill:	Expert
	Own cell phone:	Yes
	Have used IM:	Yes
	Have used GPS:	No
	Profession:	College Student (CS)

Find a friend

“It would be useful for BART. Sitting on BART for 45 minutes. It would be nice if I can walk over to the next car to find someone talk to”

First date not a blind date before actual date: “That’s a tough one. I guess I would. Unless I had a bad day and wouldn’t want it on. like mud all over me.”

Advisor: “Depending on what I was doing. If say I was going to a date. If I was commuting to work.”

Family: “Would be the same as advisor.”

“City-level keep track of friends who are far away but happen to be in town.”

Location “I wouldn’t want people to know—*he has been sitting at this location for several hours*”

Lying about location: “Probably. Case probably where I would do. I often find it useful to do work outside of either of my two offices. Some consider that would not be working.” People make the wrong assumptions

Temporary access to people: “If friends are in town…if there is a researcher I want to know…if there is a conference in town…” only people see from time to time

ongoing relationships all the time.

Active campus

All are room-level.

“I probably would only let me friends do this. Otherwise it would be useful to see whether I spend enough time in an office or in Soda Hall.”

“I probably would be more worried with my advisor or people that I work for.”

general activity: “That would be still okay with friends not with non-friends(?)”

Never get lost / Location-based searches

“If I was traveling somewhere”

Advertisements: “Preferences would be okay, but not information to help them to locate me in the future. I wouldn’t want them to correlate my requests or locations in the future with my past.” Only current location. Limit the amount of information they keep.

Mobile commerce

“I think—it would be useful. I probably would use them, but depending on how big the displays are. If it was a huge display like in MR, I wouldn’t want it broadcasting to everyone *hey looks like you’re looking for so and so* to fifty people. It’s somewhat personal.”

“Current location and body size is okay. Nothing that identifies me.”

“Just that they would store the information and everything that I have been interested in over time.”

Amazon vs. physical store: “When I do it from a physical store, I tend to have less presumption that they will know less than me. On Amazon, it’s much more obvious that they’re doing this. On my case, I am more aware of it that it happens”

Emergency

“Definitely in a case of emergency”

“I don’t think want to disclose my information. Maybe none. Then there’s no fail safe that the information will not be abused.”

Friends: “They would already have the information from previous scenarios”

Overall

Find a friend is useful. I have always wanted this for awhile. And active campus. This requires a bigger display. It allows people to keep a history.

In terms of least useful, it would be….would probably the emergency response support, because primarily, there is a lot room for abuse in this system. If this system keeps track of where you are, this could be easily used for law enforcement even if there is no emergency.

ID #2	Age:	21-25
	Gender:	Female
	Computer Skill:	Intermediate
	Own cell phone:	Yes
	Have used IM:	Yes
	Have used GPS:	No
	Profession:	College Student (Bio)
ID #3	Age:	16-20
	Gender:	Female
	Computer Skill:	Intermediate
	Own cell phone:	Yes
	Have used IM:	Yes
	Have used GPS:	No
	Profession:	College Student (Psych)
ID #4	Age:	21-25
	Gender:	Female
	Computer Skill:	Intermediate
	Own cell phone:	Yes
	Have used IM:	Yes
	Have used GPS:	Yes
	Profession:	College Student (Bio)

Note: these participants were interviewed at the same time.

Find a friend

“Now can say is there a part where you ask for my location and I *give* you my location”

“One. If I want to be found, then I want to be found. I would answer my phone.”

“If my parents would always want it on, I wouldn’t want it always on.”

“Okay *mom* okay. I would answer my cellphone.”

“For a parent, this would be a great spying tool. I just don’t like it at all.”

“If I go into a lab and I would want to find someone, she’s not there then she’s not there.”

“In the middle of the woods, if you’re hiking, if you’re dying, my car broke down.”

Active campus

Detects wireless. Doesn’t really see the point. Working in lab.

“It’s stalking, man.”

“For an undergrad it would be different. If my friends… I would be creeped if my friends found me. And they said *I saw you here*. It would just be weird.”

“for professors, it would be weird”

For meetings: “I can sort of see that.”

“I would just log off if I am not available”

“It’s just like an away message. That you’re just away. People automatically do that anyway.”

Location-based searches

“That’s really cool. That’s cool.”

“Is this a secure network?”

“It’s the same thing for the GPS system in car. It’s so useful especially for restaurants.”

“I guess the thing I use the most is mapquest.com. It’s like how to get to this place and that place. I am always looking for restaurants. If I am out with friends, I don’t know where it is.”

“Instead of the car, it’s online.”

Parking is useful. “To find a parking space. parking downtown. Where are the parking spots?”

“I pull a paper map. Or stop somewhere. Or call someone”

“I get lost a lot in Berkeley”

“If I make a right here, usually it works. Mapquest screws up a lot.”

Yes they have to prepare beforehand. Print out directions and hopefully not leave them in the house.

“would they be calling at home or spamming my mail? …If they’re interested for research, then it would be fine. if it’s spamming me…then it’s bad. Like I am a big fan of Google, their ads are helpful. It depends on how the information is used. Like on Google, and it would show me where I can buy.”

“For example, trying to find Sirius. Google found me.”

Mobile commerce

“Do we get to choose the vendors”

“I like Chinese food and Chinese restaurants spam me.”

“They would have to ask me for information.”

"I only get things I want….have option to cancel"

Gasps of surprise if ads sent to pdas.

"What's wrong with having a flyer in a section that I can look at. Why would it be different?"

based off on behavior/interests

"I don't care what other people got really."

"It's more of a turn-off for clothing. It's a girl thing. If everybody else got it, I don't know why I would want it."

"CDs I could see. Could get introduced to new artists."

"I don't need a billboard talking to me. I want to be able to walk without being bothered. Overload after awhile How do I turn it off"

"I don't like people telling me to do things. If I do things, I go ahead and do it"

Search engines: "kind of what Borders does. It's totally cool with me."

"I know what I like. I know what I want. If something is trying to figure out what I want, I don't think they would know."

"I want tools to find what I want. Not have a tool find things for me."

"I walk into shoes. I would rather do a search myself. Find all shoes all my size. Rather than something shot at me."

Emergency Response Support

"It's a good idea in a fire. It's a good thing."

identity vs. identity w/ health records

“No, gets into too much issues. too much ethnical issues. too much privacy issues”

dependent on technology

“Motion sensors could be better”

“The only thing that could be determined. Like Star Trek, how many bodies there are.”

“how does that work in a fire?”

-----------!!!!!

“people who know will volunteer….”

People know for specific locations. Work within a few meters.

“It all depends on the phone”

“Secure location where it won’t be accessible…..”

“Somebody you trust then with information.”

“It would be useful. If they can get down, then it would be useful. Then it would be useful for people in emergencies who have no idea where they are.”

Overall

Least liked: find a friend, active campus, mobile commerce least liked

Most liked: location-based search/never get lost, emergency response support

ID #5	Age:	21-25
	Gender:	Female
	Computer Skill:	Intermediate
	Own cell phone:	Yes
	Have used IM:	Yes
	Have used GPS:	Yes
	Profession:	College Student (Comp Lit / Playwriting)

Find a friend

Bookmark. Block would only happen if not blocked. Would not find application useful.

“If I was a parent or nervous boyfriend or nervous girlfriend, it would be useful.”

“in a emergency situation, if it something would happen to somebody and you could find them immediately.”

“but wouldn’t you just call them?”

father would be interested.

would let friends, not family.

“Family is already very close to you, so if they’re checking up on you…sort of already smothering and this is one step further. their intentions for discovering where you wouldn’t be…good intentions.”

“I left my keys at home. And I couldn’t get home. We just kept trying to call each other and trying to meet each other [roommates].”

Only useful for finding each other.

Useful for locality.

“good for high school student or college student. For a place of business.”

Had felt like it was a satellite tracking.

Would only be useful for one side of the find-a-friends. Bosses, especially.

“useful for a relationship that requires constant interaction”

City level would be useless.

“if you had a meeting place, general activity would be useful.”

“Nobody can say no in this society. It’s easier to be avoidant.” reference to away messages. Not putting “true” away message up on buddy list.

Active Campus

“I wouldn’t carry something like that. Would be useful to go to the city. And you can’t hear people on the fall such as the club. No signal. That would be handful, but girls would never carry. You could never put it in the pocket, and it could be easily stolen.”

Using it online might be useful.

Soccer moms would be useful. “They would eat it up. They would have it in their navigators. Suvs. I can see it right now.”

“Safeway? Are you going to Safeway? I am going to Safeway.”

“Are you picking up the kids? Am I closer by or are you closer by?”

bought in idea of lying. Social issues. Drinking at bar, working out, leave it there.

Away messages…no point of having general location, because it’s the same as location.

Location is more useful. “Extra buttons you have to press.”

Navigation

Really likes it

“411 is so expensive. And they might give you the wrong number. And you have to pay for a refund. This is so much easier.”

Movie theaters, restaurants, fabric stores, shoe stores.

“If you’re in a mall you don’t normally go to, and there are always new stores popping up. This would be so much easier.”

“I know Berkeley really well, because I get lost. You know that part of Tilden Park, I have it all in here [mind]. If you want to go from Los Angeles to Tilden Park, you can ask me.”

“I had to call my sister for directions. My doctor called me and she gave me step by step directions. Park car across the street and walk across the street. if the cell phone could know, it would be so useful.”

No for ads or give out personal information. “10% off is just like tip.”

“Only good for the tweenies really.”

Mobile Commerce

“Sales would be a fabulous addition…Stores never release information like that. Like the day of. Express. 20% off.”

“The retail girls during a sale can’t help you out.”

“With books, extremely useful. with clothes, it could be on hold or someone could be just bought.”

“Turn into an *enemy of state* thing.”

Hopefully not a minority.

In store okay. “It’s more specific. Sometimes advertisements are helpful. Sometimes you’re watching Tv and you’re like oh…1000 anytime minutes. And you’re pushing through junk mail and if you had the time look through. But the more specified and the less you have to disregard, It’s useful. If it’s tailored and you don’t have to fish, it’s all going to be helpful.”

Emergency Response Support

That would be very helpful. It’s necessary. Experienced kidnapping? useful for sending out information. Car dies.

“Make it a state requirement that the location of everybody in a building. Especially buildings on a fault line.”

“horrible intuitive feeling”

Required in offices, large questions.

Overall

Emergency response support most useful. Find a friend useful.

Never get lost the favorite.

Does not like to talk people.

“See in high school. Foresee high school problems. You didn’t add me!”

“TAs would probably hate it.”

“I am surprised that palm pilots don’t have it [never get lost]”

ID #6	Age:	21-25
	Gender:	Male
	Computer Skill:	Intermediate
	Own cell phone:	No
	Have used IM:	Yes
	Have used GPS:	No
	Profession:	College Student (EECS)

Find a friend

Finding mostly friends, people who are nearby. Questions about homework or projects. Friends are nearby.

“I don’t know if you want your friends to know where you are. There are some people who you don’t want to know where you are. Maybe friends and girlfriend, but not family.”

Coworkers might be also good. Class project definitely. Very useful. Especially when the deadline is useful. It can be very important. I don’t need to know where everybody is all the time. I suppose if they have a cell phone then you can call them where you are. Except that they might be in an area with the cellphone off where they don't want to be disturbed.

It would depend on who you're looking for. It would depend on different levels. Like certain people could find exactly where you are. Like for parents, they would only look what city you're at. You have some level of control.

There is a level of difference between exact location and *where* you are. At which school. But you could be in Soda Hall and if you're a TA, you could be there for work. Or if you're a student, you're there for school. Or you may be in the lounge or something for a group meeting. or just hanging out there. It's not necessarily where you are. It's what you're doing.

Would use invisible mode when don't want to be disturbed.

Active Campus

Not exactly where located in a room.

If I had a cellphone, then yes. It would have to fit on something small. I don't have a cellphone and I don't like carrying extra things around. That's one thing extra. It's one extra thing to keep track of. *Did I bring my cellphone with me? Did I turn it off*?

You could be in the same location, but be doing different things.

I would use an away message. Why not just use an away message instead of putting a location? More text to put in.

Location-based searches

Maybe restaurants. Maybe shops. Video game stores. Someone you haven't been before. Public transportation areas like bus stops or BART stations. It could be useful in an emergency such as a hospital. It could be useful.

Does not get lost very often, but I know people that do.

If I was going somewhere, I would print something from Yahoo. It is kind of static. It would be nice to have something dynamic. The yahoo maps are structured and there might be a situation where you get stuck somewhere. What if you drive more than 1.3 miles and then you don't know how to turn around.

If I got lost, I would ask people. If you're pretty close to where you're going, you can ask people. Or if you're totally lost, you can backtrack. However, it probably is not as good. I know people who get lost even with a map from Yahoo.

I don't know if I want people to give me more advertisements. I think it's okay if they know that they know I like Mexican food. And it would be useful if say the Mexican restaurants pop up at the top. But it's not always a good idea to have Mexican food always pop up wherever you go. If there was some way to sort. To sort by location. To sort by my preferences. Or…*I am feeling lucky*.

coupons: It depends on how intrusive the coupons are. If it rings. If I can turn it on and off, then it would be useful. Of course, it might become just like spam. Too much to sort through.

Mobile commerce

Very noisy advertisements. “You can’t ignore it. It’s just right there. But that’s the point of advertisements like that, so that you can’t ignore it.”

In the movie, it seemed more annoying. It’s just like pop ups in Internet Explorer.

Maybe you might want to go into the store and look around. And you might find something you weren’t looking for.

past purchases: It would be okay. I wouldn’t want to get more advertisements than I am right now. If they are more personalized, then they might be more effective. Advertisements would not be better not by having more since it might bother people but if it was more personalized.

You can always ask somebody for something. It’s two levels.

Emergency Response Support

Emergency 911 would be useful. Cell phone tracking would be useful.

Knowing where people are in a building.

“You can’t know where everyone is. Because not everyone is going to have one. You can’t just get the people with one out and think it’s all fine.”

Assuming that proper authorities have the ability to turn on cell phone tracking. But at the same time, other people can then get it. “You want to be prepared. You don’t want to be unprepared.”

If if you broadcast the stuff beforehand, if you don't dial 911…they are not going to know it's an emergency. If you don't dial 911, then they won't know and you would be stranded for one or two hours. If you go on a hike, then you could fall and break your leg. If you're in a situation where you can't dial 911, then it's not useful. If you have a cell phone and you can dial 911, then why broadcast the information in the first place.

Overall

Some aspects of emergency response support useful and least useful. Never get lost. In general, in every day life, never get lost is useful. Find something that you're looking for.

ID #7	Age:	21-25
	Gender:	Male
	Computer Skill:	Expert
	Own cell phone:	No
	Have used IM:	Yes
	Have used GPS:	No
	Profession:	College Student (CS)

Find a friend

Mothers are paranoid. "They want to know where you are."

"It's more of a privacy thing. I don't think people would want to share where they are all the time. I am doing my own thing. I don't want others to know"

"It would be good for new people, because you don't know their convenient. But for friends, you know their schedule. It would be useful for meeting up with people"

"There should be an option of different level of location. If I was meeting up with someone, I would want to know which building."

"Not just a building, but if you're meeting someone in a crowded area. It would be useful if it was highly accurate—walk five meters this way or that way."

"General location would be better. It would be more a privacy thing. You don't know exactly which location they are. You just need to know if they're having coffee."

"Invisible mode. Say if I was looking for another job, and I don't want my boss to know. Hypothetically if I was cheating on a girlfriend. Invisible mode implies that you're doing something bad and you don't want people to know. The word should be changed to such as *offline*"

"On instant messenger, I block people. I don't want to see them online anyway."

Active campus

Close friends. I don't want professor or TAs to know. No coworkers/bosses. They will know if you're goofing off.

If I was a TA, I would share with the professor but not my students.

How accurate the general activity would be. Because I may be watching tennis instead of playing tennis.

It’s something you would want to turn off and on. you don’t want to be forced.

I would want certain people to have less specificity.

Some people are always *be right back*. What is the point of an away message if you’re not going to be specific about it.

Location-based searches / Never get lost

Restaurants, bookstores, general hang-out places

I get lost all the time driving to a place I haven’t been to before. I use mapquest to print out driving directions. I would never some job site for an interview. I want to know the exact route.

When I get lost, I backtrack, make U-turn. I am always using mapquest. If I get lost, I would have ask someone for directions. That would suck. I don’t have a map in my car.

I wouldn’t want to share information and get junk mail. All that stuff ends up collected by companies. They in turn send you junk advertising. That’s what happens. It’s nice but I wouldn’t want that happen.

Share shopping preferences.

If they could guarantee that the information would not be used for marketing purposes, then it would be okay. Personally, I am weary of those freebies, because they are never really free.

Targeted advertisements are what you are looking for. You might find that thing that they are advertising.

Mobile Commerce

Really liked the “who bought”

“It seems like you have to agree to this one and you have to agree to that one.

I don’t buy into advertisements. It won’t be effective. It will be more of an annoyance. If I walk into a store and they know I like this product.

If you go somewhere, usually you know what you’re looking for. I am not going to a store and not know what I am there for. I browse when I am there. I don’t go to stores to browse. Browsing is more of a side-trip.

I like that part of Amazon [other people who bought this also bought this]. Very useful.

A store only has a certain amount of stuff. There’s not a high chance that they have something that you don’t know.

If you talk about Amazon it’s such a huge customer base and they have a huge selection.

For a store, there’s no point because you can just walk around see what they have in a stock.”

Physical search would be useful. Quicken things up. Very quick. Personal preferences would be good.

All these can lead to restrictions. This would lead to an ordered list. Restricted list.

Emergency Response Support

“This is really a privacy issue then. They’ll give you an ID or something. They say they won’t use it for any other purposes. But it’s just a Big Brother.”

If it was secure, then it’s useful. Useful to locate where everyone is located within a building.

“I don’t see how a government or an organization will not come up with an excuse to use it for another purpose.”

Cell phone tracking is too specific. Phone number is almost like your id.

Everything can be abused. “All these things can be twisted in a way so that they can be used for other purposes.”

Emergency 911 is good. “You’re the one to initiate it. Cell phone tracking they would always have you in the system. Why would you need to be tracked? You would call 911 if you were kidnapped. Unless they took away the cell phone and the kidnappers threw the cell phone away.”

Will not want to disclose any information. Not necessary. A good backup. Wouldn’t want that. Too much information for others to know. Not willing to take the risk.

“It depends. You’ll never know what people will do with this information. What’s the point of the entire system if it goes down.”

“Only send information when emergency was happening. They always have information.”

“All sound good in a shallow level. There’s probably something that I don’t know about and they don’t know about. Some flaws.”

Overall

Most useful: Never get lost/Location-based search

Useful to know what restaurants/shops to be in the vicinity

least useful: Find a friend/active campus. Not a necessity. A good feature, but not a necessity.

Emergency Response Support could be useful. Saves someone’s life if it worked well.

“Lots of these wouldn’t work because Americans care about privacy. Lots of controversy over smart tags.”

Concern about the tags—similar to a barcode. Could be abused to the extent where companies would know what was bought and who bought what.

ID #8	Age:	51+
	Gender:	Male
	Computer Skill:	Expert
	Own cell phone:	No
	Have used IM:	Yes
	Have used GPS:	No
	Profession:	Software Engineer

Find a friend

“If had the device, would be useful to locate family. It would be easier to locate them in a larger area.” Same group of people. Relatives of friends.

Wouldn’t want to share with coworkers/bosses. “If it’s during a work day and we are trying to get something done, then it’s useful.” When I leave for the day, that the device is off.

Groups. Turn on and off based on time and date. “I like to know exactly [where they are located].” An exact location.

Activity would be useful to know, but location is useful.

Invisible mode – do not use, just would not use the application in that sense

Active campus

“In a work situation, the answer is yes. I want it to work as sufficiently as possible if they need to find me at work.”

“If I am out of the building, then they don’t want to find me because most likely I would be unavailable.”

Showing activity: information is not accurate based on location. not very interested as a result.

“If I am at a meeting, then I can receive messages but I can’t respond to them.”

Never get lost / Mobile commerce

Applications would be useful.

Find the “best” restaurant nearby: “How much can I trust the store so that the ratings would be accurate? How will the restaurant listings be kept up to date?”

Always have a map and will not get lost. Print a map or take a general map.

“I would be sharing information if I am certain that I will not receive information that is useless.”

“In general, advertisement is not on target. Once information comes in too large, then I want to get rid of it no matter how good it is. If I can narrow down to exactly what I want then it’s useful.”

Mobile Commerce

Somewhat useful: “If I was in the men’s section, then it would be nice to know. I like to know ahead of time what is available.”

People who bought __ also liked ___: very useful physical searches: “Exactly what I want. Would be very useful. For the J.D. Salingers, I would look in one place and not realize that it would be located somewhere else.”

personal preferences: “Things today that are grouped is very important. Useful.”

Emergency Response Support

Building response service: Would be useful. “If it’s somewhere where I go to all the time, I don’t want to constantly authorize it.”

Cell phone tracking: “I don’t think I would be interested. I wouldn’t be kidnapped.”

emergency 911: “A really good idea. There are many cases when you call 911 and they do not know where you are. For example, a driver went down the hill and could not figure where she was. She had a phone but the rescuers could not find her.”

Disclosing information: “Not very interested. The likelihood of an emergency occurring is very small. The likelihood of my emergency equipment dying is very a small. But I don’t know…I guess what I would like to have some control over how long information is kept. I want to know where everything has been for the last hour. I am sensitive to having disclose my information so that someone could find me. I wouldn’t want something to constantly profile me. Though someone may not be interested in me. I wouldn’t want someone be susceptible.”

If information could be secured: “I would feel more confidant. There’s always a way to get to information.”

Overall

Useful to least useful: active campus, Never get lost/location-based searches, find a friend, mobile commerce, emergency response

Emergency response: “I have never been in an emergency, but I don’t see myself getting into one. This one seems the most useful in some sense, but not to me.”

ID #9	Age:	21-25
	Gender:	Female
	Computer Skill:	Intermediate
	Own cell phone:	Yes
	Have used IM:	Yes
	Have used GPS:	No
	Profession:	College Student (EECS)

Find a friend

friends, coworkers, bosses

no preference

“I don’t think TAs and bosses would be finding me.”

Would only want to find a TA for office hours. Would not want to be found if was a TA.

Building level is okay. Level of location does not matter.

For different kinds of people. Homework partners.

“Have two options of knowing what I am doing and where I am at.”

Would choose to be invisible if “I am by myself” For example, reading or concentrating on my own.

Active Campus

“Very weird, because you see people moving around.”

“I guess it would be cool who would want to find me. Such as class partners.”

“I wouldn’t mind, but I don’t know how useful it would be. I don’t know if everyone wants to know where I am every second.”

Would be useful because drift between different buildings a lot.

Never Get Lost / Location-based searches

Lots of enthusiasm.

groceries, restaurants, bookstores, interesting places such as museum

“I was walking across MLK (street) and I saw a museum. And it looks interesting.”

“If I am in San Francisco, I get lost easily. I usually ask people or look at my map, a tourist map.”

“Who am I giving this information to? They can spam me with ads. It can be dangerous to have your information in a centralized database. If this database was broken into by a dangerous organization, then it could be used for malicious purposes."

"Can I search for discounts? Rather than give them the information to give me coupons."

Targeted advertisements better than spam. Do not want past activity to dictate future habits.

Mobile Commerce

Could be helpful. Good if the organization does not know identity and only knows the shopping preferences. Physical search like the search in a library using key words.

"Every time I might not want the same thing."

"I never look for new arrivals, I look at the sales section."

"Again, I don't a particular organization to know so much information."

Emergency Response Support

Worried about who would know who she is.

"Not useful. Wouldn't want to be a life-threatening place."

"You'll never know what they mean by *secure*. Proper authorities aren't really *proper*."

"If I can't talk at all and I call 911, I want them to find me anyway."

"If my boyfriend was trying to find me, then he would tell the proper authorities that I was missing. Then he could tell them about my health history."

"I guess this also depends on how close you are with other people."

Overall

Emergency Response most useful

Active campus/Find a friend least useful

"If I need find someone, then I'll go find them. Call them. Use AIM to find them. I don't need to know where everyone is located."

"It is important to me for who has access to the information."

ID #10	Age:	21-25
	Gender:	Female
	Computer Skill:	Intermediate
	Own cell phone:	Yes
	Have used IM:	Yes
	Have used GPS:	Yes
	Profession:	Researcher
ID #11	Age:	26-30
	Gender:	Female
	Computer Skill:	Intermediate
	Own cell phone:	Yes
	Have used IM:	Yes
	Have used GPS:	No
	Profession:	Graphic Designer

Find a friend

Probably friends that you're meeting up with. Sometimes have to find coworkers. Different labs and know where thye have. And it would be useful if they're coming and going. Family for example if my mom is on the way home.

Someone you must met wouldn't be on your friends list. Use it more for people I am more familiar with. Not so much for someone I just met. Weird. I wouldn't care where they were at any moment.

Seems kind of aggressive. If I wanted to meet them, then I would meet them

Family mostly, close friends, coworkers. Roommates. Girlfriends.

Abiity to screen calls. Separate groups for people

Street level for usefulness. City level would not useful. If you're a student, have it say Berkeley OH really.

Would not want to have boss find me. Barely want them to call me. That's so scary. She's just the kind that might drop in at a restaurant to say hi. That's too creepy.

Doesn't that defeat the purpose to find a purpose. If the label says café then they wouldn't know what specific café. Not such a huge difference between them. If people knew.

I would want to be invisible. I am not keen on the idea of finding people. I don't think they really need me. If they want to contact me, then they can call me.

If they knew you were at a concert, would it make a conference. Then no point of calling. It would be useful in large fairs and large concert. Useful if detailed and precise.

Wanted a kind of a honing device and finding someone. And you're telling someone you're at a corner. Or miscommunication and you're at at the wrong building or the wrong entrance.

Active Campus

:So useful at school when I was there I would have put all my roommates, classmates, and floor mates study partners. After class not everyone had a cell phone, I could open the cell phone and just look up and say omg Christina in a library and I am going to visit her. And I could have opened it and found them."

Since I work all day, not terribly useful.

Just friends and family.

If I had to input something, I wouldn't do it. I think it would be useful for a general activity (if it was automatic).

Not that useful if you're going to meet someone. And you can look at it and see that someone is close by. And just knowing where people are.

It's just too similar to the map and doesn't seem anymore useful. I think I like visual representation much better. The distance better.

Never get lost

markets, gas stations "hard to find gas stations" parking structures, opening parking spaces

I need a map pretty often. Before mapquest, I would get easily lost.

When I get lost, I usually drive around. I usually use a map.

I really like Amazon's suggestions.

Information not private. Would try to sell it to people.

I wouldn't like to add a coupon. I would advertisements on my cell phone and pda. Especially if they cut airtime.

Would be useful if the coupon were there. It would be like an e-mail box. Hard to look through so many open. Potentially just delete.

If I am searching for something then it would be something I like.

Mobile commerce

Worried about how things will "pop up".

Will get more irritating. When I shop, I don't like getting bothered terribly much. I think nine of ten times I'd be irritated. If it was 10% or more then yes.

Link to past purchases useful. It might be helpful. If it was done in a way that was not intrusive and more tailored

I think we are we used to this type of advertisement. Pop ups. Nothing so new.

Usually I don’t click on “something you might like”. But for certain genres such as music and books, very useful.

Physical searches: Save so much time.

Never had a situation where they wanted. For example, in a shoe store. Something specific for than physical search but if I am browsing then yes.

Emergency Response Support

No implants! for person tracking

cell phone tracking would be useful.

Definite improvement for emergency.

All of this is useful, but it depends on how accessible this information. If it’s not regulated.

I wouldn’t want anybody just to see it. Government authorities or just police.

I wouldn’t disclose healthy history and could be used in so many different ways especially since it can be used in many different ways to be detrimental ways

If someone had diabetes

If someone like me who had no affiliations beyond getting to the hospital from a fire….no different from the next person.

Overall

Emergency response support. Emergency 911. Selective tracking. The directions useful, but the other part is not as useful.

Finding people not useful. Sort of a novelty. I doubt it would be essential and if you have a phone why would you want to be find. If it's an emergency…

Information disclosure. People find out more and more about you. Health. Not insurance company. Only the doctor.

Keep privacy when you have a cell phone anywhere. If someone can find you using find a friend, not totally useful.

I think there would be a pressure for find a friend. Like friendster someone asks people, and it's hard to say no because you can't say no because you would be mean. And then all these things come with it

to be your friend and you can't say no because it's mid and there is a pressure and you get along list of people you don't want to be friends. and you might regret it later on and you have to be put yourself on invisible. then just another hassle about it

Big brother. Just the part where you can located your friends and they can locate you and for no other reason except to locate where you are. Can see where they are. And find people invisible. One way to think about it seems to undermine human contact. Instead of calling and talking them.

undermine the trust

trust issue. you’re not already trusting them the first person.

if I was walking in while he was checking up one, he doesn’t seem trust me already!

ID #12	Age:	51+
	Gender:	Female
	Computer Skill:	Novice
	Own cell phone:	No
	Have used IM:	No
	Have used GPS:	No
	Profession:	Registered Nurse

Find a friend

lots of hesitation. cannot understand the concept finding a friend. I probably have on a cell phone their address and call them.

When people get lost, then they will lost.

Saturday get lost. First thing I go to information center and they cannot help me. I got sent to security stand and they have police over and I tell them to help me. So at that point so I could find them. People who are lost will be interested in finding me.

If cell phone can get location.

I will tell them the exact address. Then they cannot.

city is not point, because it’s so big.

General place. Very helpful. Easier for people to look for me.

Active campus

Yes, would be willing to share with people. Only people I know. Co-workers are together. Good for work so that I don’t have to over-head page.

Overhead page them. Useful for the boss. Can’t hide for myself. I am busy all the time and cannot hide myself except to go to the bathroom.

General activity would be useful. Don’t’ have to tell coworkers that is going to break. Employer is useful. Employee is okay, because busy.

Probably co-workers don’t like it when they want to hide themselves and people will know where they are.

Never get lost

If I have an appointment with a friend, so I will look for map. I will search for the library. For the bank.

If I get lost, I will go to the police. I will ask people and look around. I get lost at least once month.

That would be useful.

Coupons would be useful. yes.

Mobile commerce

I don’t like the advertisements, because it’s too personal. If I go to Safeway, then they will thank you. I don’t like it when they say my name at Safeway. Thank you, Mrs. Ng. They know my information. Still who knows. I don’t leak to too much information for personal security. The coupons after the receipts is okay.

It’s too personal. I like to look for things myself.

like recommendations, I don’t want them to know so much. Because maybe I don’t buy it, then it’s embarrassing. And to the salesman also.

Linking past purchases. It’s okay to know. Then I will get some idea of what to buy. That’s okay.

Physical searches. And if it fits my style. Then I will buy it. And if there’s a sale.

Grouping and searching.

Emergency Response Support

Building emergency response service. Very positive. Very confused. Then they can survive. It’s a good idea.

I will use cell phone tracking. That’s how they find me.

Emergency 911 If I want police to find me.

I would be willing to tell them my name, my gender, what kind of help I need.

If there was a device, helpful for Alzheimer’s disease, because they forget. Then the police can track them. Only certain diseases. Useful for kids. Kidnapped does not know and the police does not. Health-risk patients such as diabetes. And if they have seizure disorder. I think it’s very useful. Especially for elderly. If one has Alzheimer’s, the family would inform the police and they can track the woman down. There’s a lot of people run away. I have seen many in the hospital. Especially in an emergency.

Overall

Emergency response support most useful. Find a friend/active campus very useful. Very useful for employers especially to find the people who don’t.

Never get lost. Not useful. It’s easier to ask someone for a location. People know things better. Say if I want to know Starbucks then they can point out.

ID #13	Age:	51+
	Gender:	Male
	Computer Skill:	Intermediate
	Own cell phone:	Yes
	Have used IM:	No
	Have used GPS:	Yes
	Profession:	Lawyer

Find a friend

my kids. would not find coworkers. my wife maybe.

my wife would want to find me. my kids probably would find me.

Have address, not a street or city. If they are not at work or home, then I would like to know the address.

“If I am in court, then I need to have my electronic machinery off. Then when I am in a courtroom, I need to have it off.”

“It would be very unusual to be invisible. I personally don’t have any need to be invisible.”

Active campus

In a new place. In a large public place such as a football game.

“If I was at an away game working, then I would want to know where my daughter is. If I see that she is moving away then maybe someone has her.”

the activyt: “I don’t see why they would care. A small part of what I tell other people is what I tell you. Who really cares what a friend is doing. I don’t think I need to know what my friend is doing when I call him or her. I just want to hear their voice.”

“Is there a code for *getting in trouble*?”

“Maybe he goes to the tennis courts to meet his buddies and drink beer.”

Never get lost / Location-based searches

Once I find my primary location, then I want to find parking. Parking garages.

I need directions once or twice a week. So I use mapquest.

So when I go back to Washington D.C., I use mapquest to get my way from airport and hotel.

“If I get lost: call. stop and ask for directions. Sometimes I stop and buy a map.

“Who has my information?”

“Maybe for someone else. At my age, I like to pick out my own. I have a few favorites. For me, it would just be clutter. I don’t want all to know any restaurants in town.”

“I would like to have the ability to do it. I don’t want it to be displayed on my screen when I turned it on. I used it once or twice on mapquest. I don’t want it all the time, but it would be nice to have it.”

“There was a system similar to the coupon system around here. I didn’t like it. Often than not, people our age tend to look for restaurants ourselves. I don’t see much use for coupons.”

Mobile Commerce

Entire scenario does not seem that helpful at all.

“I would hate that.”

“I am sure you heard about how mean and women shop differently. Men go in and are direction-oriented. Women like to take their time.”

“I don’t want to be told that there some new jeans for me. Not very useful.”

When a man buys a new shirt that he needs to buy a tie and pants. But a man does not buy that many suits in a year.

“Would this replace salespeople? Sometimes they give me opinions. Sometimes they provide me a selection. Sometimes my wife is with me and she gives me recommendations. It matters what my wife suggests.”

“I would like access to the physical search before I left my home. If it was a remote device or whether it was my laptop, it would be great to tell what the inventory was before I left the house.”

Past purchases: “It would be okay for shopping for myself. What if I shop for my wife or my children. Then it would be all clutter.”

“Safeway collects all this data. Everybody likes chicken, but nobody likes pork. So they can base on their inventory based on past purchases. Do people buy peaches in the summer and winter. I only see it as being useful for the store.”

“Stores are not that big.” No reason to have such physical search.

“If it was a huge like a Wal-Mart and there aren’t enough clerks. So the first time going to a huge store, it would be useful to have this kind of automation.”

Emergency Response Support

More important for knowing where children under 18 are located.

Cell phone tracking very useful.

Emergency 911 very useful. It has to be done.

“I think it’s an invasion of privacy. I should have the ability to call emergency services, but I don’t want them to know of my whereabouts 24/7. I agree with the idea. If a fire truck drives up to the street and they hit a screen, and they could tell that there are four adults, one is over 70.”

“To have them my name, my income, my address, just like a credit card. Invasion of privacy.”

“I am concerned about the invasion of privacy. If it is a short-range detection device, then it’s fine. If they can tell down in some central computer room, then I am in your house or you are in my house. If they’re driving by and there was a fire here and you and I are here. But if they know downtown, then it’s a 100 feet.”

“I don’t want people in Martinez for example to know where we are located. Proper authorities shouldn’t have access to it unless in short range.”

Overall

Mobile commerce – least useful.

Find a friend/active campus/emergency response – very useful

find a friend/active campus in the context of a parent and a young child

Only short-range for emergency response support.

Location-based searches is my fourth.

ID #14	Age:	51+
	Gender:	Male
	Computer Skill:	Intermediate
	Own cell phone:	No
	Have used IM:	No
	Have used GPS:	No
	Profession:	Scientist

Find a friend

Very useful.

Would want to find somebody I know. Maybe coworkers.

Relatives, coworkers would want to find me.

Detail would be helpful. Building would be very useful.

“There’s no point in knowing where someone is. If he’s at work, he’s at work.”

on invisibility: “If I don’t want to go to the board meeting, mostly because I have relatives home. So I want to be conveniently invisible.

Active Campus

Useful, but gives too much information away.

General activity not useful. Only useful if it’s very general such as *out to lunch.*

“If they look and see that he is been sitting there for 20 minutes. It’s too much information. At work. Sometimes they can see that he has been on the computer too long. You could be straining your muscle.”

“if that person has the ability to control the general activity, then it would be okay.”

Never get lost / location-based search

Would be useful.

“if I am taking public transportation, where is the nearest BART station. Parking lot. Bus stop.”

“Every time that I go to a new place, I get lost. I ask people on the street.”

“I don’t know about giving my phone number and address to a department store. I think that will be too much information. The information is useful when you’re lost. It does concern me with all the extra information you or you may not need.”

Mobile commerce

Targeted advertisements would be useful.

General search. General idea of finding a product, but not highly influenced by past purchases.

Emergency Response Support

Would be useful.

“This is life and death, then I should. So absolutely, I would be very anxious to disclose information. If I am at work, I would give the exact location and how to get to me. The routes. Exact information as possible. When you deal with life or death, then it does not cost any more extra time to release any information.”

“With health problems such as diabetes and Alzheimer’s, then you are not aware of your own health. heart attack can strike at any time for people at a certain age even if they think they’re healthy. If they are publicly have health problems such overweight and on being on a certain medication, then they are really at risk. I think this kind of information [disclosure] is very useful.

Overall

Very useful – emergency response support

least useful – find a friend, active campus – but it's similar to emergency response support.

"If you're in the Middle East, you'll never know whether you're going to live or die. There could be car bombs or a suicide bomber. Situations like that could really influence on whether people really want this kind of information in a cell phone or pda."

"it's something that is going to happen on a cell phone or pda. They could have the location pinpointed for whatever reason. If they are in a situation where they are risking their life. The time will come when you can locate a person by a cellphone or a PDA."

ID #15	Age:	51+
	Gender:	Male
	Computer Skill:	Intermediate
	Own cell phone:	Yes
	Have used IM:	No
	Have used GPS:	No
	Profession:	CEO

ID #16	Age:	46-50
	Gender:	Female
	Computer Skill:	Intermediate
	Own cell phone:	Yes
	Have used IM:	No
	Have used GPS:	No
	Profession:	Accountant

Note: these participants were interviewed at the same time.

Find a friend

would want to find co-workers, children

“They want to find me, because perhaps there was something urgent going in the company. If they can get to talk to me, it’s already good enough. I don’t think they need to know where I am located. If you’re talking about tracking the location, then there’s a lot of privacy issues. Not many people are used to it. Then parents are consistently concerned.”

“if they are underage. Like under 15. Anybody over 15, we should let them take care of themselves.”

“If it’s work-related, then I don’t need to necessarily to know where they are located. With exception,. I have two drivers. So if I want one of my driver to pick up something. Instead of trying to contact him and ask *where are you!* Then if I see that he is in Fremont, then I can tell him to go to Mountain View.”

Street and that’s it. No need to have city and address. No point in having a city.

“For example, if I am in a store. Then I tell my daughter to stay at a store. Then I want to be able to come back and see her there.”

“Again it would be depend on the situation. I think the general place is sufficient. With the exception of the children.”

“If you lose a child, then you care about their exact location. But in general purposes, the exact location is affecting the privacy of the person.”

Usually it depends on the situation. If I am talking to a stranger, then I don’t want them to know where I am located.

Invisibility is useful. Or you can block people. Rather than turn it off unless for the children. If they are under 15, then they go off to a party, then the parent will have to know exactly where they are located. Then after the party, then I put it back on a normal mode. Otherwise, I don’t think that kind of product would be welcomed by the general public.

Active campus

No. Not useful. General activity not useful, because other people don’t really care. Too much invasion of privacy. I don’t want to be watched. I don’t want to be visible to other people, even your friends. It’s too much going on.

“Perhaps knowing the general activity would be useful for prisoners. For security reasons. For ordinary people, it’s none of their business.”

Never get lost / Location-based system

Past purchases don’t always determine future purchases. I don’t always have the same shopping preferences.

Go into detail. Too much work for tourist.

Use maps. If get lost, park the car in the street. Usually we go to a gas station. Usually find a freeway.

Whether people can afford this kind of device.

“I would rather have the option of choice. Give me a list of restaurants. Let me to Indian restaurants. Mainly because many people are afraid that they are away of other people. Privacy issues.”

“We already participated in the No Call List. Most people will be similar like me that we dislike being interrupted by unnecessary solicitation. For example, for me I get more than 500 e-mail. That kind of interruption is a lot. Even with my spam filter, it will not go through. Again coming back, we want to have a choice to not be disturbed.”

Mobile Commerce

Targeted advertisements not useful.

“It depends on the people. The psychology of the customer. Usually the customer does not want to be disturbed. However, when they want to buy something, then they want to be helped.. If there’s too much help, then that might scare them away.”

“Until the habits of the customer will change, then they will be scared. Window-shopping.”

“When I want to go to Circuit City and I look at a TV. And then somebody asks me if I want to buy it and now I am thrown away. I go to Frys and nobody is helping me. I am more likely to buy it.”

“I think it [mobile commerce] is good, but it’s more like a scheme. But you’re watching TV and all of the sudden, advertisements come up. Making it entertaining. Instead not trying to be too aggressive…or direct approach. Would shock me.”

If they know the name, what else would they also know?

Physical searches would be useful.

“Sometimes after I buy a certain item and I walk down an aisle, then I walk down and discover a new item. If there’s a terminal then I can figure out what you carry and then I can start asking questions.”

“I don’t want someone to know my favorites. It need some kind of password so that it’s reviewed by me. Like other people like ladies, you might not want other people to know your size.”

Past purchases ordering search results would be useful.

Emergency Response Support

“What I am afraid is the government will know too much. It may be abused or misused by the government. When you say emergency, how often do you encounter that? It seems to be very useful at an urgency. Emergencies don't happen daily. It may not be a good idea. I personally wouldn't want them to know."

"If the person is over 18 an adult, can the police have the rights to search with court approval? Because if the police with any other authority can monitor, then it not much of a difference to collect evidence for an investigation."

"With the proper control or approval for other authorities for example from a court, then it is okay."

Emergency 911 is okay, because they choose to call in an emergency. Give consent by calling them.

"Each person should have a choice. You have a choice to give out how much information when you dial 911. In case of emergency, you give out some kind of medical history. The amount of medical history would already be set. They might know my history of disease such as diabetes. It's already predetermined ahead of time. If I do dial, then I have a choice. In an emergency, then I may forget all my history. With the approval ahead of time, then that information is okay to give out."

"If I want to disclose something, then I would do it ahead of time."

Overall

15

- useful: never get lost/location based search, emergency response, mobile commerce with some privacy
- least useful: find a friend, active campus

“When there is an emergency, with a limited and with our own choice of information that can be disclosed, instead of the authority wanting to everything about us automatically.”

16

- useful: emergency 911, find a friend
- least useful: anything about advertising.
  - I like how you can search for your own things, but not the targeted advertisements.

Use technology correctly to enhance life. It is important that people have a choice in how much information can be disclosed, then the technology is useful. Everybody has to be cautious.

“Eventually if the government dictate what we do, then we don’t a freedom of choice.”

ID #17	Age:	21-25
	Gender:	Female
	Computer Skill:	Intermediate
	Own cell phone:	Yes
	Have used IM:	Yes
	Have used GPS:	Yes
	Profession:	College Student (Math / Economics)

Find a friend

“If you get lost, then you need some way of contacting people. What happens if there is bad reception?”

“I don’t want everybody to know where I am. For security purposes.”

“You know me, but you don’t know where I live. Then I would give exact location.”

“If they are close friends, the building is fine.”

“City would be someone who just met. Someone who is not that close. Family would be given street or address.”

Invisibility when want to be isolated.

Active campus

Easy to find. “If I have a friend who walks really slow, then I can figure out how long it will take for her to get to a location. Then I can approximate the time. It is more efficient in a way.”

“I don’t think I need the general activity. Why do I want people to know that I am studying?”

“If you send a message to someone, then it would be useful in knowing why they are not responding me. Like *okay she says hi to me and just left?*”

Never get lost / Location-based search

All aspects would be useful, except giving out personal information.

Not really. It depends on what you need at that time. It depends on the situations.

“if I have a map, then it’s easier to find directions. If I can’t find some place, then I’ll figure it out. I am a lazy people so I always have to ask people.”

“I don’t really want to share my personal information like my address. But I don’t mind sharing my personal preferences such as past purchases.”

Minimize my budget. Save money. What if I change my mind that day.

Mobile commerce

Might be useful. Physical search be useful.

I would want my specific location. Then why go to a store then?

Emergency response support

Emergency 911 very useful. Cell phone tracking for only at-risk people. Building response service only in those high-risk situations.

Only in an emergency then they can find me. If I am in a fire, then that’s the only time they should be able to locate me.

“Do they watch you every minute?”

Depends on the situation, if you’re in a security situation. I don’t see why it’s necessary. I would only use it only emergency cases or when it’s necessary to locate people.

“Use it when in emergency or at night.”

cell phone tracking: “For separate age group. For the really old and really young. For disabled people.”

“Would only disclose information if only proper authorities had the access.”

Overall

most useful – emergency response service

least useful – advertising specifically mobile commerce. Compared to others not important. It would cut down shopping time, but it doesn’t seem significant.

ID #18	Age:	16-20
	Gender:	Female
	Computer Skill:	Intermediate
	Own cell phone:	No
	Have used IM:	Yes
	Have used GPS:	No
	Profession:	High school student
ID #19	Age:	51+
	Gender:	Male
	Computer Skill:	Expert
	Own cell phone:	Yes
	Have used IM:	Yes
	Have used GPS:	Yes
	Profession:	Systems Engineer

Note: these participants were interviewed at the same time.

Find a friend

Would be interested in finding family, friends.

Finding coworkers. Currently use the cell phone a lot to contact coworkers to find out what is happening.

Parents. Having boss finding. Concerned about how the project is going.

Computer industry directly connected to customers. “They really need to contact me. Cell phone.”

“Only family and friends. Maybe classmates. Friends who want to hang out with me.”

See son turn off cell phone when he does not found.

Level of location for different people.

“People I didn’t know where I was…exact address. Be weird.”

“If I know who that person is, then I would give them the exact address.”

“Use cell phone a lot for business and sometimes I would like to know what’s going on.”

Do not want to have high school teachers find her.

Useful for certain people for the label of location. Helpful only for the people they know.

Want to be invisible when don’t want to talk to someone. Maybe for parents. Social thing.

“Not useful for invisible, because they would not want me.”

For business off-hours, I should make this functionality be available. So that co-workers can reach me.

Active campus

Participants seem very impressed. More like a novelty device rather than something useful.

high school student: Don’t really want people. “What if I tell people something different and they notice.”

systems engineer: don’t want to show where I am all the time

Context aware location would be useful. “Like an away message?”

So you know why people are not responding.

Never get lost / Location-based computing

Very useful. Want to find friend’s houses.

Find restaurants, hotels in a new, unfamiliar location. Airports.

high school student: Don’t drive, so don’t get lost. We look at the map. Ask people.

systems engineer: local. out-of-town. very easy to get lost. If I get a car with navigation system, then I ask for directions.

GPS system much more useful. Not as easy to ask people. Sometimes they don’t know and they give a longer route and they confuse you even more.

“If you go there frequently…that would be cool!” for relating past purchases

very useful

Advertisements. “Awesome. That would be cool.”

Mobile Commerce

Would be useful. I don’t like spending time to shop. I just go there and get what I need. I don’t know what I want to get, so it affects what I want.

“When I go shopping, it’s such a hassle to find the right size. Just annoying times”

Don’t have to spend time looking. Very useful.

Go to a store, but don’t know what you want.

Suggestions are helpful. Input from other people. That [certain clothes] will look good with the particular clothes.

Search engine would be useful.

“It would speed up shopping process.”

tailoring search results

“I don’t go toward a particular brand. I don’t usually conform to a brand. Past purchases don’t define my future purchases. It wouldn’t that helpful, but it would be cool.”

If things are narrowed down, then be very helpful.

Emergency Response Support

building response – very useful

cell phone tracking – very useful. only for those kidnapping situations. not just because parents want to find me Parent talks about how he wants to install a black box so that he can track how fast his son is going.

emergency 911 – very useful.

to give extraneous information - commercial purposes, don’t like it. “Depends on how much I can control. If I don’t know that party well, then they can use it for some commercial. third party dealer.”

Overall

most useful app for high school student – find a friend, active campus

most useful app for systems developer – gps device because can go to a new city and be prepared

least useful app for high school student – mobile commerce, the least important. it would be useful. it’s a bonus but not essential.

least useful app for systems developer – find a friend/active campus not as useful. no need to locate people, but just need to communicate with them

like how to contact everybody in daily life. “Make daily life much easier.”

social issues – another side effect would be too much information go out. Maybe the information will get into the wrong hands. All of this. Privacy

“But I am an optimistic man, so the technology outweighs the social issues for me.”

Some of the features are really cool. I don’t want people I don’t know to know exactly where I am. If it gets in the wrong hands, it can be scary sometimes.

Useful if driving and the car is breaking down in the middle of nowhere. More utility.

I wouldn’t want my parents to know exactly where I am. I like the features of it. I want my own freedom. I want my own freedom.

Invasion of privacy. Big brother.

ID #20	Age:	21-25
	Gender:	Male
	Computer Skill:	Expert
	Own cell phone:	Yes
	Have used IM:	Yes
	Have used GPS:	No
	Profession:	Free lance web designer

Find a friend

Would want to find friends. “I would use it for spy work and find out if my brother was up to no good. Then I would track him down.”

“My parents are usually at home or work. I don’t need to find them. I see my brothers often enough that I don’t need to find them.”

“No I don’t think I want to share my location. I hide a lot and I don’t want people to find me.”

“They can call me on a cell phone if they want to find me.”

“For my girlfriend, I would share it some of the time. She would get suspicious. She would use it all the time.”

“I think it would be cool to change the level of information whenever I wanted. So if I can set it settings that are more specific when I want people to find me. But I pretty comfortable with having everybody knowing what city I am in. Street and building I would like to control.”

“Can manipulate the general location to trick people. Don’t think it’s a great idea.”

Active Campus

Would not want to use it. Even with trusted people, not that willing to use such a device. When going to a location, can be doing many different things. Any activity determined from the context will not be accurate.

“That’s scary. Anybody can just bomb a small area and you will die.”

Location-based search / Never Get Lost

restaurants, banks, parks, internet cafes, stores depending on need

“Normally have a map beforehand. Usually find my way.”

“Usually call who I am visiting for directions. Last resort, ask somebody walking the street.”

“Computers try to be useful, but usually end up being pain if they personalize things.”

“Advertisements bombarding my cell phone. We already have enough advertisements from TV and radio.”

Mobile commerce

Not very fond of the idea of mobile commerce

“The whole fun shopping is to go in and find what you need. Not have everything presented to you.”

Physical search engine would be useful. However, would rather look at physical items.

Customers who bought this CD bought this: Want control to turn it on and off. “Not useful. It’s everywhere.”

personalized search results: “Only if you can turn it on and off at will.” Usually buy things for other people 25% of the time

Emergency Response Service

Highly dependent on how easy the device would be to use.

Building response service: Probably use it. Different from *find a friend*, because it’s only for emergency purposes.

Cellphone tracking: “If everybody knew there was tracking on the cell phone, then the kidnapper would know about the tracking and throw that out.”

Emergency 911: Very useful.

Disclosing information: Okay to do it, because nothing to hide. “Some people might be embarrassed disclosing such information. if the people who handle the information can be taught to treat the information responsibly. Then have them to sign a legal agreement that the information cannot be used for anything except emergency purposes.”

Overall

Most useful: Emergency Response Service The ability to find nearby businesses in Location-based service

Least useful: mobile commerce, advertisements of Never Get Lost/Location-based service, find a friend/active campus not useful, not many people would use it

Specific advertisements are the only ones that are relevant. “I pay attention to only 1 out of 100,000 ads.”

“I have a certain friend who likes tech stuff and they would like this. I have another group of friends who would be freak out by this. They wouldn’t like it. Privacy.”

Only useful in college.

“The same thing that happened with radio, then with TV, with computers, then will happen to handheld devices. When radio came on, the only purpose of radio is to sell radio. Advertisements. The Internet used to be free 10 years ago and there’s money-making schemes everywhere. Then it will go into cell phones. I know the cell phone industry is suffering. If they do this like mobile commerce, I think they're selling out!"

Appendix C – Freeform Survey Comments

This appendix contains all of the freeform comments from the survey conducted by Scott Lederer [96]. These freeform comments were used to inform my analysis of end-user privacy needs, as presented in Chapter 2 (Section 2.2.4) of this dissertation.

These comments were collected by Lederer, but have not been published before and are presented here courtesy of him. To make the division of labor clear, Lederer collected and analyzed much of the numerical data in the survey, while I did a further analysis on the freeform comments.

Every row in the table below is a response from a single individual to a given question. An entirely empty row means no comments were provided. In some cases, email addresses were included in the freeform comments. We took care in obscuring these email addresses, replacing, for example, `person@somewhere.com` with `xxxxx@xxxxxx.com`.

Do you think "faces" is a good term to use for this purpose? Choose the term you think would be best: {face, mask, role, profile, other}	**Which is more important in deciding your preferred privacy? {situation, recipient}**	**Freeform comments**
		I think you have violated the premise of your study by sending out your spam email.
Something on the front of my head. Or a character in the A-Team.	Realistically, I'd rarely remember to switch profiles between activities. Much easier to set up a filter based on remote id. And as far as importance, well, I'll tell people I don't distrust what I'm doing or have done, so the situation isn't relevant.	
A configuration for the amount of personal data disclosed in response to a query from a third party.	The person receiving this is information is the key factor, since people who know where I am and other personal data some of the time are likely to have a good idea of the values for the same data at all other times as well. Of course, "boss" and other people with whom my relationship is very strictly constrained by context should get responses dependent on my current context.	I sure hope that if yall are implementing this, the "Blank" face is default and it is made _very_ clear to the user that increasing amount of information disclosed should be done carefully and with thorough forethought.
		Please let us know the results of the survey.
A face defines which of my personal information is made available to others.	Generally speaking "who" is more important although situation can also have an effect. This is because usually we do not feel comfortable sharing personal information with unknown/unfamiliar persons/organizations no matter what the situation is.	

In this context, the amount and accuracy of the personal info about me that I allow others to see.	Who receives my info is the more important one. My significant others can know where I am at any time, but for my boss, a random person on the street, and a vendor, tis none of their business. the 'who' is higher priority than the situation. Given a 'who', I then determine what that person can know about me and it rarely changes due to situation. If it does change according to situation, it's by general rules, such as that my boss doesn't need to know anything about me when I'm not at work.	"Who" is way more important than the situation. Who receives is most important. My signifcant other should see my truefacefaceface always. The evil national chains should see my blank face always. The people between these two extremes will see different faces depending on the situation. The situation is needed to disambiguate people who are not in the two extremes of intimacy.
a face is an outward persona, a collection of facts about you that you wish to let someone know.		
set of parameters defining my virtual identity		
A view of me, catered by me.		
The kind of information disclosed to another party	There seems to be very high coupling between these two abstractions, making it difficult to think about them seperately, i.e. they are not necessarily orthogonal. Instead, more orthogonal components might be "formal/informal", "romantic/friendly", "business/friendly/family" etc.	The abstraction of a "face" is useful but not complete. A "face" that I would find more useful might be "tell this person things I might want my girlfriend/mom/good-buddy/childhood-friend/future-prospect-for-a-one-night-stand to know." These might be further decomposed as described above.
A face is an artificial outward appearance.	No one should receive my personal regardless of the situation I'm in. I believe privacy is absolute.	This sounds like a reason NOT to carry a cell phone or similar device in the future.
An indication of how much of your personal information is available to the caller	I would NEVER want anyone to have that information. It's none of their business and an invasion of my privacy, even if it's a spouse. I wouldn't own a phone that does that.	It's a scary idea to even allow this information to be collected. I wouldn't trust that anything is blocked.

persona conveyed to others	both are equally important... while you might want certain people to get truefacefaceinformation at times, there are times when privacy is of utmost importance Both are critical. I want to know who is accessing my data and when, and have the final say-so if I'm going to release any but the most basic pieces of data. Also, I want to know as much about who's asking as they're getting from me.	"Who" was much more significant to me. I didn't want people in either a position of power over me (the boss) or with the ability to annoy me (marketers) to access my info. The particulars of the situation seemed less important in the above scenarios... though I can certainly envision some less innocent activities that would change that.
An online personal ad that is tied to your location and current activities		
An abstraction for representing particular aggregates of privacy parameters.		
I think of a face as a user profile - something like a "Yahoo profile". It represents an identity of the user, and the user must be able to choose different identities based on the context as well as the requester ..	The "who" is generally more important to me than the situation. I organize trust boundaries based on people and my relationships with them. So my SO will have access all the time, unless I'm unhappy with her for some reason :-) On a secondary basis, I would use situation to circumscribe my work life from my personal life. So I wouldn't want my boss to be able to contact me at some arbitrary time unless we had a prior arrangement to that effect.	Sounds pretty nice !
A profile	The person who's receiving my personal info. Context isn't that important, because I trust the person to know how to exercise discretion.	

information about me i want to broadcast ....like an eye catching slogan on a t-shirt or a homepage on the intermuhnet.	only who receives my information. what should situation have anything to do with privacy? i do nothing i'm ashamed of. if cameras were everywhere and my activities were constantly broadcast, but to a random people, then i would feel an obligation to behave in a manner that society approved of and yet no-one could take advantage of me (the way a dictator might). privacy is for criminals and other sorts of sociopaths.	Something there is that doesn't love a wall, That sends the frozen-ground-swell under it, And spills the upper boulders in the sun; And makes gaps even two can pass abreast. The work of hunters is another thing: I have come after them and made repair Wher
A face encapsulates the amount of information about you that you trust other people with.	Who receives my information is more important than the situation I am in. Either I trust someone with my information or I don't--it doesn't depend on where I am. it is more important to identify who receives my information... my situation would not matter as much	1 is more important that 2. Family comes first, followed by business, then potential business. Lastly strangers are greeted if the situation permits.
a face gives selective information based on your situation, location, and who wishes to contact you		
A face is a greeting. You use different greetings for different people in different situations.		
A degree of openness about my personal status.	Who is more important than where/when. Unless I'm cheating on my SO, I don't mind if they know exactly where I am and what I'm doing. That goes for close friends as well. I do need to know who exactly will take this information. If that information is vague, I would instinctively choose the blank face to protect myself (kinda like walking on the street in NYC. You just don't make any contact with anyone. They could be crazy. They could mug you).	I'd be impressed to see a cell phone that knew what I was doing (without having to be told). Pseudonyms aren't that important to me. If someone finds out who I am and is determined enough to get my email address/phone number/address, then so be it. It's not like the information isn't out there already.

A face is the set of information about you that is presented in a particular situation, to another person or entity.	Who receives the information is generally more important to me, in that I would prefer to limit personal information to a very small set of people. The situation is also important, however, since depending on the circumstances, I would be occassionally be willing to relinquish more information.	
..what Mephisto is to Dr. Faustus	(1), because I am not in all that many situations.	
A face is a description of what personal information one is presenting to some entity at a given point in time.	Both. The pair of the two pieces of data (who, situation) may be used to look up in a table of faces that you want to present to the world. Maybe you wouldn't want to make this table exhaustive (e.g. have some notion of groups - "work-related" people vs "friends" vs "relatives"), but I don't think it makes sense to say that either one is more important than the other; the appropriate face depends on both, and cannot be determined by only one (at least for most people/places).	Interesting - I'd like to hear about the results, when you gather/publish them. Could you contact me when they are available? xxxxxxx@xxxx
The amount of information my mobile device will present electronically to the world	Who receives my information is much more important than the situation I'm in - for instance, I would never want a reatiler to contact me unasked, but always want my spouse to find me. Business contacts might be an exception - during the work day, or after-hours during crunch time, I'd want my boss/coworkers to find my - after hours I'd rather be more anonymous	Face is as good as any - profile has some usage, though, for internet accounts. I think the way someone finds me would be a primary determinant of which face to present - if someone detects my phone nearby, for instance, I'd want to present as anonymous a face as possible; someone looking my name up from an email would get a similar amount of info; my spouse would have a username or password to get more in-depth information (like having a variety of IM handles).

A face is a filter to control the amount of information I want people to be able to gather about me.	The situation controlled my answer, but only slightly. As a general rule I don't let anyone have my personal information anonymously. The one situation where I chose not to display a blank face was the bar - I would display select information (a cross between vague and blank) to improve my chances with the ladies.	Overall, I find this idea frightening.
	ME	
It is the information you are ready to release to any other individual or organization at a given time.	The only situation I can conceive allowing anything other than a "blank" face is when my personnel secuity is seriously threatened. For example, the phone could have a panic button that immediately dials 911 and send the truefacefaceface. Or, it could monitor my breathing and if it stops dial 911. In both cases, I am the one to proactively connect and release the information - the authroities can not query it. I would need to be able to assure they could not.	Very scary thoughts for those the enjoy and protect their privacy. I prefer the old style phones. If my spouse or boss want me, they can call me. I don't care about bookstores, and I definitely don't want strangers finding things out about me unless they have the courtesy to ask me directly.
a gateway to personal information	Who receives the info is the most important. I don't want my boss to have all personal information at any time, and I certainly don't welcome strangers to my 'truefaceface" face.	

a shorthand for my "identity"--in this case, my public persona.	i would never allow want strangers to have my identity and location if at all avoidable. it's already difficult to avoid. i would only want my boss to know where i am if it is extremely beneficial for him to locate me. i would only want my husband to know precisely where i am if he were to meet me. In the case of a bar, it is easier to have that info scanned than for me to speak and provide it. if he were not invited to join, i would not have that info available. so: coarse grain scale: i'd prefer that no one but those closest to me receive ANY info about me, and for those closest to me, it depends on the situation and if i wish to be met.	i see no value in vague faces. i would only use such a system if it were the best i could do--a kind of security by obscurity or security by random noise system.
a publicly accessible profile of some sort	the who is more important. some people have priority, either because of a trust relationship (family & friends) or an economic one (boss). none trumps basic privacy for me, but then I don't pick up my home telephone when I don't feel like it, while I have friends who, amazingly, will let the phone interrupt them at dinner or sex. the when has some bearing, but it is based on the who. mosty employer-employee relationships end at 5PM, hence the blank face to the boss after hours.	Um, they interact. During the workday, I am (to some extent) willing to grant some information to my co-workers, especially if they are actively collaborating with me. Out of the workday, I am far less willing to grant much of *any* information to them. On the other hand, the "face" model--of summarizing a lot of attributes together--strikes me as slightly dubious. For example, I would have few (if any) opportunities to need to present a pseudonym to anyone who already knows me; similarly, people who know me would probably not need any of my profile. (People who don't know me should bleedin' well ASK; why are we assuming that people will tend to meet largely online?)
A face appears to be a mechanism for summarizing a series, or set, of personal information about ones activities and location.		

face - information about oneself that you particular other has readily available	who is more important. for people that know you it is a matter of setting a boundary (e.g. i dont' want to be under direct surveilance of my husband or boss no matter what i am doing). for people that i don't know it is probably more determined by situation - if i am at scientific convenction, for example, then i wouldn't mind strangers seeing truefacefaceface.	i think face is a bit too personal, while it might just reflect a principle.
my identity	I am concerned about identity fraud and do not wantanyone to get my truefacefaceinfo	this equipment to invade my privacy should be illegal
The information you willingly present to the outside world stored in your cell phone	To me, "who" is definately more important.	
A face is not an identity. It is simply the information that you wish to broadcast about yourself.	Who receives the information is definitly critical. I would say that it is never appropriate for advertisers or my boss to have any of my personal information and my spouse can have any information. The reason for the first fact is that 1) about 0.1% of advertising appeals to me in the sense that it makes me want to buy something; 2) my boss needs to trust me without having the ability to check up on me. For the case of the signicicant other, I'm sure that no amount of information that does not involve actual verbal contact with me will not suffice. Therefore, the amount of information that she could get without talking to me is irrelevent.	I have a concern. I can imagine situations where making this sort of information available to others leads to constant quizzing. For example, the boss could ask "So, where were you at around 4pm?" The significant other could ask "So, why didn't you invite ME to the show?" I can see why certain profile information would be useful to advertisers especially if they know your hobbies, etc. when you pass by one of thier data-gathering posts. However, if they can get an email address from you, why wouldn't they just email you regardless of your interests? This would be consistent with the advertisers that I deal with.

Outward appearance-- what the world sees and can surmise just by looking or "looking" at someone.	Who receives the information is more important to me in terms of my personal privacy. A stranger on the street doesn't deserve to have the ability to find out my name, e-mail address, interests, etc. just by looking it up after seeing me somewhere. Neither does a bookstore or other business. As far as the situation I'm in when the information is collected, that's not as important. I'm in that situation no matter what and it's nothing to be ashamed of or anything, so if someone finds that out, it's their own problem. :) I guess if I were someone who would cheat on my significant other or fool around during work hours, I wouldn't want that information to be public. But I don't do those things, so it doesn't matter to me.	Both are important and have interchangeable significance based on the situation. For the closest circle of family and friends, it does not matter which situation we are in - but there are nuances like surprise element or doing something secretly to have things arranged ahead of time - these elements will be lost if all details about situation are known. For people who are in secondary social/personal circles, I would rather let them have vague/undisclosed face - just because, it should remain my discretion as to what I would like for them to know at that point. If I want to provide more information, I can go ahead and allow them to see it later on.
It is the details about me that I would like others to see. A facade.		
Faces summarize levels of anonymity and privacy.	For me, they have no effect. The reason I chose "blank face" for all the situations is that the face mechanism, as described, gives me no indication when someone queries my location, name, etc. Thus, for example, this scheme could be used by a boss to constantly track an employee's location without the employee knowing. That isn't possible now because currently the boss would have to call the employee and ask his location every 5 minutes, hence the employee would catch on quickly that his privacy was being violated.	I do like the term faces, though.

A way to protect information by providing different views to different people.	(1) How much the already know. How much I trust them (how well I know them). (2) The situation matters somewhat - the bar example was good. However, most of the time I think that the person is more important. Might also consider time roles: during business I don't care if my boss knows where I am, but on the weekend / at night I would prefer she didn't know.	It depends: for my signicant other, I would always allow any information to be accessible for my boss, I would allow him/her to know where I am, but only during work related situations (work hours). for other, unkown random people/business, in general, I would not let them know anything
Similar to an online identity: people already use different email addresses for different purposes (work/friends). Only close friends would also get work email addresses.		
A set of constraints which, when applied to the set of all people, would result in a subset in which you are a member.	Both are important.	
Information about where/who I am and what my interests are.	(1) I don't mind spouse/significant others/good friends knowing where I am most of the time, but I would prefer most of the information is not available to strangers.	
A face is a disclosure of personal information (or perhaps more correctly, a level of such disclosure).	The situation is the most important factor, because I would use the situation to determine to whom and at what level to disclose my personal information.	In selling or describing this idea, "face" is definitely a much catchier term than the others. :-) (no figurative pun intended.)

A "face" is like an alter ego used to disguise your truefacefaceactions, except in the case of the "truefaceface" face.	It is very important as to who is looking at my information. If my spouse were looking, I have nothing to hide from them, my privacy is not an issue. However, a friend or boss does not need to know "everything" about me, and I would likely use a "vague" face. Strangers, coroporate or otherwise, have no right to know anything at all about me, and I would almost exclusively use the "blank" face.	Interesting subject matter. I never thought about cell phone usage like that before. It is very scary to think someone could ever even get the point of being able to "track" me. Although I said I would a "truefaceface" face with my spouse, I don't like the idea of ANYONE being able to track me like an animal. It seems "Big Brother" really isn't far away, based on this survey. Very scary thought indeed.
what others see or know about you	who receives the information is more important than the situation. your relationship with that person will determine how the situation is perceived, therefore, the person who is looking up information about you is more important than the situation in which the information is collected Who is most important to me. I don't feel like I keep secrets about particular things I do from anyone unless I would keep everything secret from that person.	Who receives is by far more impoarant than what is received. The only thing that a spouse gets out of this is not having to let you know they are inquiring about your location and activity. This is usually given though a phone call or SMS anyway, so it's nothing new. The situation is imporatnt as well if someone is doing something that requires discression, but changing your face to your boss only while you're skipping work may raise suspisions anyway. It's ok for friends to know everything about me, but is not ok for a stranger to collect my personal information anonymously. However, it is ok for anyone to approach me and have a conversation.
A face is a particular set of descriptives whose space includes name, location, email address, activity, etc.		
A face is the personal information you choose to allow some subset of individulas or entities to observe.		
My current and permanent information like location and name.		

As you have presented it, it is simply a profile of information available to others.	For me, "who" is all that matters. If I don't trust the person with personal information, I wouldn't want to give them any information at any time. If I do trust the person, I'm willing to give out information freely.	If such services were available, I'm afraid employers would start to require, or at least strongly encourage, their use. Currently, I have a great relationship with my boss, and I would always present my "truefacefaceface". But that's just >now<. In general, I expect I would present a "blank face" to my boss.
a face is a set of privacy preferences configurable relative to situation or data collector.	who receives my information is more important than the situation in which my data is collected. particularly with the cases involving those with close ties to me, i'd lack plausible deniability if i was wearing anything less than my 'truefaceface' face towards my significant other while out with my friends. similarly, my boss would probably wonder why i was blank during office hours if i were paged at work.	it was difficult to fit the privacy elements of faces as described into some of the situations. for example, my friends and employer will always know my truefacefacename, and a stranger in a bar will know where i am and what i'm doing.
An aspect of yourself that you project onto others	"Who" is more important. For example, my signifcant other or complete stranger always gets the same information, no matter what situation. For others in between the two extremes, "situation" becomes a factor, but the level of privacy still depends on the receiver. I think that the person who receives the information is far more important than the situation. The problem with giving away personal information is that it's difficult to undo that operation. My primary concern will be thus be the people who can gain access to my information regardless of the situation; only in certain cases will the situation make a noticeable difference in the information I'm willing to provide.	My answers appear to indicate that *who* receives the information is most important. However, the two situations above are informal. In a more formal setting (meeting room) I might give my boss access to my "truefaceface" face. Also, other aspects of my particular situation might color my decision. If I am mad at or cheating on my spouse, for example, she might get the "blank" face.

A face is a collection of personal information (at varying levels of specificity) that serves as an abstraction for quickly choosing which personal information you wish to reveal to others.		
A face is the role I play with regards to another individual or group.		
A "face" is an abstract representation of your "state." I.e. your identity, activity, etc.; a facet or view of information about you.	Generally, I feel who receives the information is more important that where I am. I guess I am comfortable doing whatever activities I chose to do; however, I am concerned about the prospect of who is collecting that information and what they plan to do with that information. At the minimum, I could do without more personally directed spam.	I like the current situation where I can wear a cellphone and chose whether to interact with the caller or not, depending upon what I am doing and the caller ID. This allows people to contact me if the need to, but maintains a measure of anonymity. I feel rather cynical about how my information would be used if it were being actively collected. If I started receiving spam from stores that I happened to walk past or found out my boss was checking where I was during the day, I'd probably stop carrying a cellphone.
A face is a profile of yourself that you present to other people/entities.	In general, (1) can determine everything about which face I would choose. Only when in a specific situation, like work, would something change. The recipient is more important than the context, because the information will likely outlive the circumstances (especially if retailers harvest it).	In some situations, it is more important that people know where you are and what you are doing. In some cases, like your free time with your boss, it is none of their business. Also, strangers dont need to know anything so that should be hidden. Who receives my information is more important; I don't have anything to hide from people I know, but I don't want to get spam from businesses.

An avatar or profile -- a context-dependent definition of what others may know about me.		
A face is the image you choose to present to whomever is looking		
A set of attributes that one can change somewhat arbitrarily in order to control the amount of information others have about you.		
A "face" is a set of kinds of information about a person. Some kinds of information might vary over time.	Who receives the information matters. The situation does not matter. If I were to changing faces depending on the situation I was in, I would feel like I was playing some sort of cat-and-mouse game, which would be very unnatural and uncomfortable.	You missed a whole class of faces: False faces. Even honest people might want to present false information to someone while planning a surprise for them. Other subtleties that might matter to some people: When someone looks up my information, do I get a log of that event? Do I get information about them? I might be more willing to give more information when people are willing to reciprocate, rather than watch me surreptitiously. I might be willing to give out more information if the query is flagged as an emergency and comes from someone I trust not to abuse that flag.
	who is more important	
Ur (possibly inaccurate) identity visible to others depending on policy u set	who receives it much more important situation matters rarely	

As being used in your survey, it is one's electronic identity presented by the cell phone to others and is defined by a particular combination of information parameters.	Both, equally, and most important--try actual human connections to regulate information.	You should also be asking exactly how secure you think this proposed system is or is not.
But my words are warped by those of Goffman!! In your system, a face is a facets in my world.. Basically, it stands for an aspect of my identity that i will present given a particular situation. Who i am given contextual information of people, place and time. [I would not use the word face for this but i understand your use of it/]	Both of these contextual cues are essential and can be broken down further. For example, some data i can always assume that certain people are welcome to know (such as my wife will always know my truefacefacename). Other is situationally dependent... For exampl	Face has connotations with one's expressions and emotional state. It is a presentation with a lot more depth than simply the role that you are playing or the facet of your identity that you are showing... Of course, this is a long rant that should probably be discussed over the phone because i've been a flaky friend and still haven't responded to your last 5 messages... Sorry about that...
A level of information sharing. A privacy threshhhold.	Who recieves the info is far more important. Anyone who's trying to sell me something I don't want or need (as oppposed to something I've gone out to buy) gets no information. Sharing a little bit of information with random strangers is fine, but I'd rather _not_ share with retailers than share with strangers. I think it would be nice to be able to not disclose my location/activity sometimes, but mostly I don't care if my friends/so know where I am and what I'm doing.	who receives the info is more important than the situation. Certainly truefacefacein the situation presented in this study
a profile		
A view of your personal information, with a configurable level of detail	"Who" is much more important. I do not wish to allow strangers or advertisers to contact me or collect information about me without my explicit approval	I am wary of trusting my cell phone to protect the security of any personal information.

A profile of a person and their current agenda.	I believe the first factor is the most important... No matter what the situation, I would always present the same information to the same people.	
A face is your complete external profile available to third parties. It should be configurable.	Who is much more important that the situation b/c every device is a primary device for either work or home and you think about a device as such.	Nope - thanks
The level of information that you are willing to allow others to see at a given time. You are able to control the amount of information by choosing the face.	Who receives the information is much more important than the situation. I dont mind some people having access to all of my information including where I am, while I dont want most people to have any access to my private information (name, location, etc.). The situation is of little consequence unless you are trying to hide where you are.	
a privacy profile	Who receives the profile is the primary determining factor. I imagine that for each of the parties described in these scenarios, I would have a few custom "faces" that I would use almost exclusively. My situation (time of day, activity) would define which specific face is shown. So the "who" does the most narrowing down... Then the "situation" determines what is shown to the requestor.	
A self composed packet of personal identity.... Sort of a "mask" only in this case the masks are removed...	1 - who. Even then the situations are vague, and one could, realisitcally lie about them. (eg. in scenario one, you COULD be having a steamy lunch with a female intern) But then I wan't big on anyone knowing anything about what was happening in my life.	I like face, as a term, but I think that in this age of AOL-ers profile has a wider understanding.

The level of personal information that different people can access at any given time.	For me, who receives personal information is more important than the situation I am in when the information is collected. I am generally always honest with my significant other about where I am and what I am doing, whereas I feel that my personal life is none of my boss's business, but I'd feel more comfortable disclosing some information to him or her than to a complete stranger, who should not be able to learn anything about me without my permission. The physical situation I am in, however, is more or less public information to anyone who might see me, so unless I was doing something I was trying to keep a secret or felt ashamed of , I wouldn't care who obtained that info.	This kind of "checking up" on people isn't possible, is it?
It is a preset group of privacy settings that you would be able to choose based on the current situation you're in, or who is looking at your information.	I think situation is more important, although who receives it is pretty important too. There are some people who I wouldn't mind letting know where I'm at regardless of what I'm doing (i.e. spouse), and some that I wouldn't want getting my information at all (like a store). But, there are some people who I wouldn't mind knowing if I'm at home, but I might mind knowing if I was at a bar during the working day (like a boss). Also, I probably don't want random strangers approaching me if I'm grocery shopping, but if I was in a social setting, it might be a good way to get to meet new people.	It would be great if the faces were customizable or automatic. For example, it would be neat if I could program the phone to always put on the "blank" face if I was at a bar or a location I don't want to advertise. It would also be cool if you could set up accounts for users, such as if your significant other signs in, they can always view your truefacefaceface even if you have your phone set to vague or blank. That way, you could do all the customization at once and not accidentally forget to change your face if you go someplace where you want to keep your privacy. One other general comment I have is that even with the blank face, this system still doesn't completely ensure privacy. People might assume that if I'm blank, I'm doing something I shouldn't be.

A persona	(1) is more important than (2)	I think the WHO is more important than the SITUATION. More over, if it's people you don't already trust, then I don't want to give MY information to them. I'd rather process the information THEY give me. The store should give me the specials and my device will process my interests for me... I only want to broadcast to the ones I trust...
a predefined set of information about myself that I'm willing to give out		
a face is a filter that transmits a certain subset of information requested by an entity who wishes to know something about you. the face can differ depending upon what you're doing, where you are, and who is asking for information.	i feel that who receives the information is the more important factor. i don't tend to do embarrasing things so it's really more of a matter of who i feel has a right to ask things about me and enforcing those opinions.	scott lederer rocks! you go girl!
basic information about yourself	the recepient is more important beacuse getting no information instead of specific information is information itself.	
A combination of who I am (name and contact info) and what I'm doing, and where.	It's mostly a matter of whether I think it's any of their business, with a side note of what they already know. There's no point in hiding my real name from my boss, for example. My primary partner gets to know most of what I'm doing and where--it's how we've chosen to relate. Other friends are likely to know my name and email address, but maybe not where I am. Sometimes "I'm busy" is all that's anyone's business. None of this is any of a store's business unless I'm asking them to deliver books and send a confirmation email.	Can we have an option to pre-program location/activity information? Not just "busy" but, for example, a preprogrammed "at work" that will turn up regardless of my, or the phone's, current physical location.

A face is the information you present to the world. It seems that it is information that you would be comfortable giving the person if they asked you in conversation.	I think the two are fundamentally linked. I think that with some people, close friends and sig others, I rarely have situations I'm in that I don't want them to know about. However my boss can know about what I'm doing when it's work related but not what I'm doing in my free time. At work you also don't want random people bothering you so the place matters too. And frequently I don't want random strangers to know everything about me without my gettig to at least communicate with them on some fundamental level.	Could you make your face lie? Could it tell your boss you were at work when you weren't. How does the face know what you are doing? B/c if you can make it lie then it seems like there would be little point.
a level of privacy	who is all that matters to me; I would not alter based on the situation UNLESS my boss required to know where I am (and then I would, of course, only be OK with this if his queries were during business hours and relevant to a work related issue)	Who is important with close family. They can know anything, anytime. Who is also important with retailers. I don't mind if a company knows 'vaguely' about me as long as the spamming effects are completely tailored to what I like. If there's a great French restaurant around the corner ... absolutely I want to know about it. But for most other cases, situation is most important. If I am on the job, then I probably want to hide from friends but I have nothing to hide from my boss or other coworkers. If I am on my own time, then the opposite is truefaceface. If it were just myself and a spouse, I wouldn't want to be bothered by anyone :)
an intentional, constructed portrayal of self		

Information from which an "impression" may be formed. e.g. a stranger's first impression	I do not feel that strangers should have any access to personally identifying information about me. The only reason I answered that my spouse/significant other should have vague access is because it is sometimes convenient to know my general location rather than having to call me and bother me. My spouse/significant other obviously knows my number, but even my spouse/significant other should not have the ability to track every precise move that I make. Only 911 should have this ability so that I can be located by public servants in emergency situations.	Good luck. I have a healthly level of paranoia, but sometimes I don't think I'm paranoid enough. It's not pleasant to feel constantly under a microscope even though I'm an upstanding citizen.
A face is the set of information you, in essence, publish about yourself by having your phone with you and turned on, etc.	Situation is not a consideration, because if it was, then observers would be able to distinguish "private" situations from "public" situations, which would presumably undermine the goal of using situation as a criteria. So one should attempt to decide what face to present independent of situation. On the other hand, the "who" is all-important. Very close friends (esp. spouse) are already trusted with private information, except usually precise info about activity. Strangers are both opportunity and danger; opportunity for doing something or meeting someone you might not otherwise, but danger of having your information collected systematically and exploited. So above, for strangers, I tried to only expose info which might be useful to the potential acquaintance but close to useless for systematic collection purposes.	I've actually given these ideas quite a bit of thought before your study. There are certain people I'd like to be more available to, and others less available. It would be absolutely critical for the device to be clear about what information was being made available, and to whom. I'm not sure whether the device should log queries; if queries are logged, they might be interpreted as a social gesture, which in turn would make them less likely to be used in some cirumstances. I also think a law would be needed to explicitly protect against collection and exploitation of this kind of information (this law is needed already). Technical measures alone are not sufficient (despite engineers' chronic blindness to this fact), and the market never protects privacy.

The information you reveal about yourself to a specific source at a specific time or during a specific activity.	In all the cases above, the face I would use is independent of the situation, and is completely dependent on who is receiving the information. I figure if I want to hide any of my activities from a person, then I would always want to do so. Otherwise, the person would be suspicious whenever they can't access certain information. I think that the appropriate level of privacy is much more dependent on who is receiving the personal information that the situation I'm in when the information is collected. The relationships that I establish with individuals (or companies, in the examples above) tend to transcend the activities in which I am engaged; once I choose to trust someone with my information, it's less important to me to be able to change it moment to moment than to maintain and protect that information consistently. The person who would receive my information matters to me much more than the situation. The situation matters in some cases; for example, if I were browsing in a bookstore I'd be much more willing to let the bookstore look up my information to present me with a special ad than I would be otherwise. Also, if my boss were a good friend of mine I'd be much more willing to let him or her know more information, but perhaps still not during working hours! Aside from some cases like this, though, I would want to decide how much information to give to any given person/entity and not change it too much from situation to situation.	(1) - there are people that I do not want to know who I am, or how to contact me. situation has very little impact. Who is receiving the information is more important because I should know who my personal information is going to.

My "faces" are the representations of myself that I expose to other individuals. These vary, depending on what I'm doing, where I am, and who the other individuals are. I use my "faces" to keep my personal life private, maintain plausible deniability, and protect my personal information in ways that I choose.		
A "face" is the degree of information I wish to allow certain others to access about me at any given time.		
personal profile. some form of id, contact info and basic description of me.		
A face is a momentary identity and status report that varies in detail.		

The information that someone can access about you	"Who" is more important than "situation". If someone has access to your info sometimes, then he can infer or deduce things from being denied access at other times, so modifying "face" based on what situation you're in would not be an effective way to protect privacy Who recieves my personal information is far more important than the situation. My boss -never- gets to know where I am outside of work without my knowledge, even if I'm doing something I know he/she would approve of, or doesn't care about. Nor do random strangers. For reference, I chose "vague" to the "spouse or significant other" questions not because I wouldn't provide him with that information, but because it's going through a computer and is, as such, inherantly insecure -- and so much personal information combined with my exact location is -not- something I want going through one insecure channel at one time. Besides, if he really needs to know where I am, he could call me.	1) this is possibly the most important: no one I don't know is getting that kind of information about me if I can help it. 2) This is less important, but still matters... my boss, say, has no business knowing what I'm doing when I'm not on the clock. I think it is very important that random people cannot have access to your personal information. If it is your spouse, of course they should be able to know where you are because you shouldn't have anything to hide from them. If is your boss, it's O.K. for them to know to a certain extent. If it just says you are at a certain location, your boss may make assumptions that are not accurate about what you are doing. This could be dangerous. The only people who should have access to your information are people who you know and approve of having the info. It is perhaps more important to take into account the situation you are in, though. Just because, wrong assumptions may be made judging on where you are.
A "face" represents the level of information I am willing to give out to a given person.		
A face is in this context is the sort of information that one chooses to present to the world and the members thereof.		

A face is what people see when they look at you. Not only what you look like, but what they can infer about your personality and interests by your body language, actions, company, etc.		
The information (or lack of information) that you send out about yourself, which varies depending on who is requesting the information and what you are doing at that particular moment	Who receives the information is definitely more important. I don't tend to want people to know anymore about me than they already do. This applies to both the "who" and the "situation". For example, my friends should always be able to see my truefacefacename and primary email address because they already know that, but depending on what I am doing, I may or may not want them to know what I'm doing or where I am. If I am not available, I would want to be able to leave an away message as in IM.	I like this idea very much. I use IM all the time when I'm home, and I would love to be able to constantly be "sending out" my status (where I am, whether I'm busy), but I definitely want control over what gets sent to whom and when.
It's a profile containing basic information about the owner of the "face."	For the most part, who receives it is more important to me than where I might be found at any given time. I gave blank faces to spouse/significant other because I feel really strongly that partners should never ever be checking up on one another. I feel like that about bosses as well, but I do recognize that while you are on the clock the boss may have some right to know a bit about where you are and what you're doing. I gave a truefacefaceface to the stranger while making a large assumption, that this was a stranger I was interested in getting to know, thereby encouraging me to provide as much access to me as possible.	It was interesting. Can you send me results? xxxxxx@xxxxxx.com (truefacefacee-mail address - LOL)

A front	Work people can know my information during work hours. Home/SO people can know my information always, though not to the point of keeping tabs. Random people might have access to enough information to help start up a conversation, but nothing beyond that. Random businesses should never get any personal-indentifying information (vague might be ok if the business can't figure out who I am - though I'd be skeptical).	Thought provoking. Nice job, Scott.
an appearance, a collection of perceived qualities	They can both be important. Nobody needs to know where i am and what i am doing, really. Except my husband. I deserve at least that much privacy. But I wouldn't mind allowing my family and friends to have a vague idea where i am during certain times that i choose. that could be handy. but at other times unnecessarily intrusive. the same goes for companies collecting marketing information. there are certain instances and certain companies where and from whom i appreciate targeted marketing	Will you publish the results of your study anywhere? I would like to see the results! Please let me know - xxxxxx@xxxxxx.net Thanks
The ability to show the information that you chose from your cell phone.	I feel that we need to have the choice of who and when we want to give out our information.	there doesn't need to be any restrictions on who recieves your personal information. a coffee shop owner is no differnet then a government employee. The situation you are in when the information is collected is a little more sensitive. I do no feel it is important for anyone to know you exact location at all times.
a "face" a profile similiar to what you would find already on the many chat and instant messaging services out there.		

A limited set of personal information whose depth and precision I would like to be able to tailor according to the identity of the inquirer.	I care much more *who* receives my personal information than the situation in which it is requested. In certain cases, information about my current activity should be restricted depending on the inquirer, but in no situation would I want personal information released to anyone but a limited set of people I've selected.	"Face" suggests it is my identity which is changing. I prefer "mode" as it more accurately reflects the different phases of my daily activity - for example, during the day I might choose to be in "professional mode," in which I consider my coworkers, boss, clients, etc have a right to locate me. At other times, I may be in a "recreational mode", in which only friends have access to me. Even further, I may choose to be in "private mode", in which access to me is extremely limited. While I do not think I would ever want to receive solicitations from retailers, I can imagine a "shopping mode" in which limited information is released.
a way of managing personal data in a seemingly intuitive way. a way in which others perceive you (which you can manipulate) a way of managing interactions between/across virtual and real environments, and between very different kinds of parties (known individuals, unknown individuals, consumer bodies) in different kinds of relationships with the 'user'	i think i would also be concerned about how i am generating and updating/maintaining my personal information ... i am more concerned about who receives personal info ...	how is activity level determined? could you lie? and it would be fascinating to see if there were difference by gender and age or ethnicity here -- are you tracking that? (this would be F/35/Hillsboro/australian)
A collection of information about me making up a composite "image" for someone else to look at.	For me, who receives the personal information is more important -- because if it's a party that I know and trust, the situation doesn't matter (i.e. I don't mind them knowing what I'm doing/where), and if it isn't a party that I know and trust, then I don't want them knowing anything at all about me.	

a mask... some way you can hide yourself.	mainly, it depends on who, with the exception of some special (bad?) situation.	this "facing" is a very impressive idea. i'm wondering how, in practice, often one change her face, or think about changing it.
A level of detail with respect to personal information about me.	"Who" is most important. I can use my degree of trust in people to determine what they should know about me. Situation is less important. Perhaps if I'm in a setting where I know the interest in my information is professional (a conference, meeting) or familial (family reunion, family visits). Basically, if there are strangers around, and the situation doesn't clue me in to why they would be interested in information about me, I don't give them any of that information.	
It's a form of social interaction. It communicates to the observer something about the person who wears that face, whether or not that communication contains any truth is a different issue.	I think the most important factor is who receives the information. I, personally, do not want to share my moment to moment activities to anyone in the ways described above. Situation, however, focuses on something an individual has control over, so if you are doing something indiscreet that's a personal choice, however, sometimes you just don't want to talk to some people and you have no control over who wants to reach you, so I think that is the more important issue.	
The way that you are presented to the world.	In my case, I'd never, ever use that feature of my cell phone. I don't want ANYONE to know who I am, what I'm doing, or my "interests" - friend or stranger or marketer. If someone knows me personally, they should know me well enough by my human interaction with them - not from some electronic informer. And the rest of the world - mind yer own business.	I like "face" but I think "profile" might be a bit more clear in the beginning, since people are used to using "profile" for systems like AOL. "Face" would probably give it that new special twist that marketers like, though, so I went with "face" in my choice above.

A face is one way of deciding just how much technology is able to invade your private life.	"Who" is always important. I am perfectly happy going through life without the people I meet on the street even knowing that I exist, much less how to contact me or what I do in my spare time. "Situation" is only important at a few select times and for a few select people. For example, I would hate to have my girlfriend see that my location is "flower shop" when I'm trying to surprise her, but I would be willing to let my colleagues know where I am if we are trying to meet to discuss work.	
Information that you make available to a person depending on their relationship to you. Similar to having a 'work' email address based on your legal name, and a corresponding signature with phone number, but another one based on your nickname set to sign with a quotation about your hobby.	My relationship to the person retrieving my personal information is more significant to me then the situation in which it is collected - The location where I am when someone searches to find my occupation doesn't make as much of a difference to me as a random business being able to retrieve my business email address just because I passed by their store.	
A profile of personal information.	Who receives the information is much more important than the situation. The recipient determines my level of trust not to abuse my personal information. The situation is only relevant if this trust is limited enough to reveal more information during certain activities (e.g. boss finds out about work-related activities, but not social ones).	

My face is the information that would allow someone to address (contact me, talk to me, etc.) me if they looked me up using particular software. Using my truefacefaceface, people can address me using my real name and they can believe the information they receive. On the other hand, my vague face allows people to address me using some kind of name, but they have false information about me. If I give my pseudonym as "daffy duck", then they would know to doubt the rest of the information. They could address me, but they kind of know that I want to hide some things.	The 'who' of the matter is more important. I don't mind if my friends and family know where I am and what I am up to. I want to reserve information from other kinds of acquaintances, and I never want strangers to know. I just can't think of an activity where I would want a stranger to know more about me. Maybe I wouldn't mind getting offers from a resturant, but I wouldn't want to give out my personal information.	I don't know if 'profile' is the best term. It may be because I'm a geek and that's what I'm use to. Also, I feel like there is a big difference between tangible information about myself, such as my name, address, etc, and more transient things like my activities and interests.
set of information about yourself that you select based on what information you want people to know about you.	Who is receiving the info is more important than the situation. Who the person is defines my relationship with them. The level of trust is determined by the relationship, the possible motives they have for finding me etc... The situation can be relevant if it is relevant to the relationship, for example I would not have a problem with a boss seeing my "truefaceface" face during working hours but during my lunch and on the weekends, a boss has no right to this information.	

A "face," at least in this context, is a cell-phone user profile which gives an interested party certain information (name, location, email, etc.) as chosen by the cell-phone user.	I think in general (1) is more important; it doesn't bother me now, for example, if people I know but have fallen out of touch with try to look up information about me online, or if a random stranger IMs me out of the blue because my profile suggests we share common interests. I'm much crankier when large businesses try to get information about me in order to market their products at me, particularly when said marketing involves spam. I'd be a little leery of a program that tells *anyone* my current location/situation, though.	
A 'face' is whatever a person is showing to other people, that is how they act and behave in front of other people.	1) I prefer onlt people I know well to have my contact info, and for the most part for strangers to leave me alone unless I want to get to know them or we ahve a mutual friend in common that brings us together. 2) If I feel that the situation I am in is friendly and that I will like the people I will get to know, then I am all for it. If I feel threatened or disliked, I do not want these people to get to know me.	
Dynamic Information. So far, information about somebody was static. For example address, email, and phone number doesn't change dynamically. I think "face" is the first kind of information about somebody which changes dynamically but still is truefaceface. Actual location and actual activity continuously change.		

A "face" is an image of myself at that instant that I want the world to see.	(1) is definitely important. For eg. if my significant other ever looks up my info, it should provide maximum information. (2) is important because I dont want someone to bug me if I am on a vacation.	
a persona	It's very important WHO gets the information. Someone as close as your spouse should theoretically know your information anyways. Someone who doesn't know you at all shouldn't have your information. Situation is important, but not as important. The right to privacy should not be interfered with.	
Personal info such as name, age, address, etc. Location, activities, etc.	Both are equally important.	Don't like the idea of "tracking" movements and disclosing identity. It's a slippery slope and therefore I prefer to make no exceptions based on who is looking for the info. Of course there may be emergency situations where it would be helpful to a significant other. However, I'd rather not disclose any info of the sort described above.
current known identity.	Both are equally important to me. While I am working, I expect a limited level of privacy with those associated with my work. I expect to recieve marketing information because, indirectly, that is apart of my responsibilities (regardless of my position). Perhaps a company can offer a solution or item better than one I currently utilize. While I am not working, I do not give a damn about my boss, others, or marketing materials. I expect complete privacy.	I understand the term 'face.' The term 'privacy,' however, seems more appropriate. For instance, 'my privacy [level] is X' is more understandable than 'my face [value] is X.' In this instance, 'face' is vague and its value is not instantly ascertainable. 'Privacy' is clear and directly related to its purpose.

a mask and a personal information filter	who recives my information is most important. changing the amount of information reavled under different circumstances is information itself. if don't reveal to you friends where you are exactly does that mean you are on a date? savy and smart people will be able to extrapolate information easily.	i like the option of the other face. for the most part i don't stay with options given.
The information made available to various people	The situation is more important b/c i would want to be able to limit what people can find out about me	