Diss. ETH Nr. 16100

Personal Privacy in Ubiquitous Computing

Tools and System Support

A dissertation submitted to the
SWISS FEDERAL INSTITUTE OF TECHNOLOGY ZURICH

for the degree of
Doctor of Sciences

presented by
Marc Langheinrich
Diplom-Informatiker, University of Bielefeld
born February 25, 1971
citizen of Germany

accepted on the recommendation of
Prof. Dr. Friedemann Mattern, examiner
Prof. Dr. Günter Müller, co-examiner

2005

Abstract

Visions of future computing environments involve integrating tiny microelectronic processors and sensors into everyday objects in order to make them "smart." Smart things can explore their environment, communicate with other smart things, and interact with humans, therefore helping users to cope with their tasks in new, intuitive ways. However, this digitization of our everyday lives will not only allow computers to better "understand" our actions and goals, but also allow others to inspect and search such electronic records, potentially creating a comprehensive surveillance network of unprecedented scale.

How should these developments affect our notion of privacy, our "right to be let alone," our freedom to determine for ourselves when, how, and to what extend information about us is communicated to others? Should we give up our solitude and anonymity in light of these new technological realities and create a "transparent society," in which nothing can be kept secret anymore, for better or for worse? Or do we need to surround ourselves with better security mechanisms that will make our communications and our presence untraceable to anyone but the most determined observer?

This thesis argues for a third alternative, a middle ground between the two extremes of abandoning privacy and attempting full-scale anonymity. It proposes an architecture to facilitate the upfront notices of data collections in future computer environments, means to automatically process such announcements and individually configure the available collection parameters, processes to store and subsequently process any such collected data automatically according to the given notices, and tools for individuals to control and inspect their state of privacy in an ever connected world.

In particular, this thesis provides for

- a method to announce privacy policies in smart environments via *privacy beacons* and personal *privacy assistants*,

- a method to reason and act upon such policies by automatically configuring the available services with the help of *privacy proxies*, and
- a method to store the collected information and enforce their respective collection and usage policies through *privacy-aware databases*.

Taken together, these mechanisms can provide the technical foundations for future privacy frameworks that provide a level of privacy protection suitable for smart environments: anytime, anywhere, effortless privacy.

Kurzfassung

In zukünftigen computerisierten Umgebungen werden winzige Mikroprozessoren und -sensoren in Alltagsgegenstände integriert sein, um diese „smart“ zu machen. Smarte Dinge können ihre Umgebung wahrnehmen, mit anderen smarten Dingen kommunizieren und mit Menschen interagieren, um so ihre Benutzer beim Bewältigen ihrer Aufgaben auf neue, intuitive Art und Weise zu unterstützen. Diese Digitalisierung unseres Alltags wird allerdings nicht nur Computer dazu befähigen, unsere Handlungen und Ziele immer besser zu verstehen, sondern ebenso unseren Mitmenschen ermöglichen, diese elektronischen Datenspuren zu durchsuchen und damit potentiell ein flächendeckendes Überwachungsnetz von Orwell'schen Ausmaßen Realität werden zu lassen.

Wie sollen diese Entwicklungen unser Verständnis von Privatheit beeinflussen? Werden wir gezwungen, unsere heutige Form der Privatspähre angesichts des technisch Machbaren aufzugeben und eine transparente Gesellschaft zu erschaffen, in der es keine Heimlichkeiten mehr geben wird? Oder müssen wir uns umso stärker um verbesserte Sicherheitsmechanismen bemühen, die es uns erlauben, unsere Kommunikation für Fremde unhörbar und unsere Anwesenheit unsichtbar zu machen?

Diese Arbeit schlägt eine dritte Alternative vor, einen Mittelweg zwischen diesen beiden Extremen von totaler Transparenz und absoluter Geheimhaltung und Anonymität. Sie stellt eine Architektur vor, die den frühzeitigen Austausch von Datenschutzregeln in zukünftigen computerisierten Umgebungen gestattet, die automatische Verarbeitung solcher maschinenlesbarer Ankündigungen zur individuellen Konfiguration der verfügbaren Dienste durchührt, und die die datenschutzgerechte Verwendung der dabei ausgetauschten personenbezogenen Informationen ermöglicht. Gleichermaßen wird den Benutzern ein Werkzeug zur Verfügung gestellt, mit dem sie den aktuellen Zustand ihrer Privatheit – wer hat wann und wie lange welche Informationen über mich und zu welchem Zweck gesammelt – zu jedem Zeitpunkt feststellen und gege-

benenfalls korrigieren können.

Dazu liefert die vorliegende Arbeit die folgenden Beiträge:

- eine Methode, um maschinenlesbare Datenschutzregeln in zukünftigen computerisierten Umgebungen durch *Privacy Beacons* automatisch zugänglich zu machen,
- eine Methode, um in Abhängigkeit dieser Regeln und aufgrund persönlicher Präferenzen mit Hilfe von *Privacy Proxies* Entscheidungen zu treffen und eine Dienstumgebung individuell zu konfigurieren, sowie
- eine Methode, um die so erhobenen Daten im Rahmen der angegebenen Regeln in einer unterstützenden Datenbank (einer sogenannten *Privacy-Aware Database*) zu speichern und zu verarbeiten.

Zusammengenommen können diese Mechanismen eine Grundlage für zukünftige Datenschutzsysteme bilden, die einer Umgebung voller „smarter“ Gegenstände angemessen sind: Datenschutz überall, jederzeit und ohne größeren Aufwand für den Einzelnen.

Contents

Preface

The essence of civilized life is sharing space with others without intruding or being intruded upon.
Barrie B. Greenbie[1]

With the recent developments in miniature microprocessors, wireless communication technology, and low-power sensors, visions of smart environments come to mind, where computers as we know them today disappear into floors, walls, and everyday things in order to unobstrusively help us with our everyday chores. Not only can information be literally at our fingertips, a wide variety of (yet to be conceived) context-based services could free us from routine tasks and make our lifes more convenient, more efficient, and safer. A much more frightening, yet possible, vision – resulting from this technical progress, however – is that of a perfect surveillance society, where most of our lives are constantly monitored by countless digital cameras, various location technologies and ubiquitous sensor systems. A society where biometric identification mechanisms, intelligent clothing, and digital payment systems unobtrusively record mundane details of our physical, psychological, and financial health. And where comprehensive digital dossiers not only allow the inspection of all our past actions, movements, and utterances, but also provide for increasingly accurate predictions of our future behavior.

This dissertations aims at providing key elements of a technical infrastructure that helps us to protect our privacy in an age where the boundaries between reality and the digital world of computers begin to disappear. It develops these technical components based on a brief but comprehensive analysys of privacy – its history, common definitions, and legal realities – and illustrates why those who build ubiquitous computing environments will have to take extra care when designing

[1] In *Spaces: Dimensions of the Human Landscape*, Saybrook Press, 1981. As cited in [261]

"smart" computing systems. It also describes and extends the well-known OECD Fair Information Principles and discusses their implementation in the context of future computing and sensing technology, in order to make the frightening visions of an Orwellian society impossible. Last not least, this thesis puts these analyses into practice in form of a prototype-system called PawS, which allows us to complement legal and social privacy protection mechanisms with supporting, invisible technology in an age of invisible computers.

A Growing Concern

Discussions about privacy have a long history, and various historical events have brought about a change in perspective of our privacy needs. While the earliest references were mostly concerned with bodily and territorial privacy, the age of telecommunication shifted attention to communication privacy and, ultimately, information privacy.

At the beginning of the 21st century, most developed countries now have comprehensive privacy laws that provide their citizens with a reasonable balance of personal data protection, convenience, and safety. However, many of these laws have their roots in the centralistic data collection days of the 1980s and are still struggling to come to terms with the realities of the Internet and its World Wide Web. A new future full of invisible computers will require yet another adjustment to make legal realities reflect the technical and procedural realities of proactive data collections and personal surveillance devices. Also, the almost limitless level of observation and control that computers embedded in rooms or everday artifacts make possible, will require societies to reevaluate many of their long-standing norms and ethics: Should smart cars allow unfit drivers to start the engine? Should people have the ability to disclose large parts of their lives for a small discount? And how much should law enforcement agencies be able to observe the minutiae details of our lives in order to detect unlawful behavior?

The above questions illustrate what makes this new field of computer science, often called ubiquitous computing or pervasive computing, different from other domains with respect to privacy: the totality of its vision, the far reaching implications of the deployed technology, and the seamless integration of it all into our lives. Specifically, five basic properties of disappearing computers make for a qualitative difference from traditional data processing systems:

- *Collection Scale:* As computers disappear and blend into our environment, they also cover more and more areas, buildings, and things. Consequently, decisions made in the design of ubiquitous computing systems and artifacts will affect large parts (if not almost every part) of our lives, both in time and space.
- *Collection Manner:* Not only should computers be everywhere, future computing visions want them to actually disappear from our views. Consequently, deciding at what times one is interacting with (or is under surveillance by) a computing or communication system will be almost impossible in the future.
- *New Types of Data:* Next generation sensors will allow high quality audio and video feeds from cameras and microphones smaller than buttons. Even emotional aspects of our lives, such as stress, fear, or excitement, could then be sensed with reasonable accuracy by sensors embedded in our clothing or in our environment.
- *Collection Motivation:* While many of today's data collection systems aim at predicting consumer buying habits or simply provide vendors with the postal or electronic addresses of possible customers, "smart" environments can potentially make use of any type of information. As such systems will need a large pool of prior "experiences" to draw from, almost no information will ever be useless anymore.
- *Data Accessibility:* Though often not yet fully realized, ubiquitous computing systems will need to excel at data retrieval tasks in order to correctly recognize previously encountered situations, identify the current context, or allow users to efficiently sift through terabytes of multimedia recordings from personal memory amplifiers. The better these systems get at finding what they are looking for, the easier it will be to abuse this information for unintended purposes.

Providing a technical solution under such circumstances is challenging at best. However, privacy is far from a recent trend, and has thus prompted people to devise and implement mechanisms for its protection for quite some time. While the unique characteristics of today's technological advances will most likely render existing solutions impractical, it nevertheless pays to reexamine them in order to avoid obvious mistakes and achieve more efficient mechanisms.

Mechanisms for Privacy Protection

Before setting out to assemble a technical infrastructure for privacy protection, we need to take stock of the array of mechanisms available to us – both those working in our favor, and against. Building a technical tool, whether for privacy protection or other purposes, cannot be done in isolation from the legal and social realities that inevitably surround it. Otherwise we might easily run the risk of creating unworkable or inacceptable solutions.

The field of *ethics* in general, and technology assessment in particular, can provide valuable insights into the requirements and limits of any privacy solution, as it reflects the moral realities of how much or how few privacy is deemed desirable. Another important component is *trust*, since data collection systems require some basic trust in either the technology itself, the entities collecting and using the data, or law enforcement mechanisms that allow interactions with untrusted parties.

Corresponding *privacy legislation* can often help strengthening any privacy conserving system. While some basic similarities exists, legal protections differ substantially around the world. The sectorial frameworks in the US have seen a number of recent additions that specifically address issues such as location privacy, while European law with its more comprehensive protection still requires corresponding updates that take into account the recent technological developments.

Of the existing technical solutions, maybe the most prominent ones are those for *encryption and authentication*. While often used synonymous with privacy tools in general, such security mechanisms cover an important part of technical privacy protection, though not the complete range of issues. *Anonymization and pseudonimization* are another building block in providing privacy when the full disclosure of one's identity is not necessary. These mechanisms are complemented with *transparency and trust* tools, such as the Web technology P3P, which allow data collectors to describe their collection policies in a machine-readable format and communicate these to their data subjects.

By being aware of the full range of mechanisms that are at work in the field of privacy – social mechanisms such as moral, ethics, and trust; legal mechanisms such as laws and regulations; and technical mechanisms for solving different distinctive problems – we can hope to build a comprehensive solution that solves the right problem, in the right manner, with the right mechanisms.

Guiding Principles

With the wealth of mechanisms in mind, we can set out to draw up a number of principles that are to guide technical development. As a starting point for such guidelines, we draw from a well-established set of practices with more than thirty years of "experience": the Fair Information Principles, drawn up in 1973 in a report by the US Department of Health, Education, and Welfare (HEW), which form the basis of practically all modern privacy laws today.

Among the most fundamental requirements is that of *notice and disclosure*: There should be, simply stated, no hidden data collections. Ubiquitous computing systems will per definition be ideally suited for covert operation and illegal surveillance, no matter how much disclosure protocols are being developed. It will always take special detection equipment to be reasonably sure that a certain room or area is not being monitored by others. But for those who want (and are bound by law) to "play it by the book," some kind of announcement system would be helpful that would allow them to openly announce otherwise covert data collections to customers, employees, and visitors, but also to family members and friends.

Given that individuals know about data collections taking place, they can exercise another fundamental requirement of data collection regulations: *choice and consent*. Again, the area of pervasive computing poses new challenges in this respect, as not even a button-click – the established means of giving consent on the Web – will be available in most of these smart environments. Users will need delegation mechanisms that allow for an automated pickup of privacy announcements and subsequent decision-making on the basis of previously established preferences.

Should an offered service be not to the user's liking (with respect to his or her privacy), a choice should exist involving *anonymity and pseudonymity*. While several anonymization schemes for Internet service access exist, their deployment in future computing scenarios is made difficult by the fact that real-world data is much harder to anonymize than virtual data. Especially the realm of location anonymity and pseudonymity would need to be part of any privacy protection scheme for ubiquitous computing.

Adequate security, i.e. encryption of electronic communication and storage, as well as authentication and access control, must of course also

be involved whenever data collection takes place, otherwise promised collection and handling practices can hardly be guaranteed. Although a large number of encryption mechanisms and security procedures exist, finding the right balance between security and usability will be a challenge for any application involving invisible computers.

Trusting a system, and especially a system as far-reaching as a ubiquitous one, requires a set of regulations that separate acceptable from unacceptable behavior, together with a reasonable mechanism for detecting violations and enforcing the penalties set forth in the rules. Technology can help implementing specific legal requirements such as *access and recourse*, so that data subjects can see for themselves what information about them is on file and potentially correct or delete it.

Even with a ubiquitous computing systems supporting all of the above requirements, situations may arise where getting the explicit consent from a subject beforehand will be difficult, if not infeasible. Complementary mechanisms such as principles of *proximity and locality* should be embedded in the underlying infrastructure in order to not only prevent accidental data collections (e.g., a memory amplifier recording without its owner being present) but also limit data dissemination (e.g., keeping sensory data stored close to its collection place).

Whether the above six points – notice and disclosure, choice and consent, anonymity and pseudonymity, adequate security, access and recourse, and proximity and locality – can be realized in future computing systems, will of course depend to a large extend on the intricate interplay between technology, social norms, and legal obligations that together will form the design space of any such environment. What we can hope to achieve is building a system that complements, rather than replaces, these mechanisms. We call this *privacy awareness*, rather than *privacy protection*, indicating that its effectiveness rests on *supporting* existing social and legal tools, not on replacing them.

PawS – A Privacy Awareness System

Total privacy is neither possible, nor desirable. Neither is total security. Our privacy awareness system (PawS) presented here follows a fundamental concept in today's democratic societies, that of the politically mature citizen.[2] Citizens are given the ability to respect other people's

[2]This is a translation of the term *Mündiger Bürger*, a concept particular to Germany, though with a universal applicability across all modern democracies. Sometimes this is also translated

safety, property, and privacy, and society relies on corresponding social norms, legal deterrence, and law enforcement to create a reasonable expectation that people will follow such rules. For example, streets and sidewalks allow pedestrians and drivers to conduct daily traffic in a reasonable secure fashion: pedestrians are expected to stay on the sidewalks while drivers are obliged to keep to the roads. No one explicitly prevents a pedestrian to suddenly step onto the street in front of a car, nor can we be sure that a particular driver will not veer onto the sidewalk and run over a pedestrian. Social values (e.g., the public shame of having killed a person), legal deterrents (e.g., fines or jail sentences), and "technical tools" (e.g., raised curbs, pedestrian crossings, driver's licenses) are in place to allow both drivers and pedestrians relative freedom of movement.

PawS similarily draws upon the concept of politically mature citizens, and tries to provide support for their daily information management. PawS is a prototype system that provides *collection* and *processing* tools that allow data collectors and processors (e.g., service providers) to communicate their collection and procession details to their data subjects (e.g., service users), and help them keep their promises. While in individual cases more protection might be required (e.g., for sensitive data such as health records), most situations of our daily life should be adequately "protected" through such tools and corresponding enforcement and recourse mechanisms that allow holding people *accountable* to their public statements and actions.

Specifically, PawS aims at supporting the following requirements: giving notice and disclosure, allowing consent and choice, and providing user access and recourse (through automated policy enforcement). Through these mechanisms, PawS provide users of ubiquitous computing environments feedback on and control over their state of privacy. PawS can be complemented with anonymization schemes such as location privacy and extended with context-specific data resolution strategies (i.e., reporting more or less accurate or timely data). It relies on standard encryption mechanisms such as SSL for communication privacy and supports signature schemes for non-reputability.

PawS consists of four parts: one or more privacy proxies that handle all data exchange, a personal privacy assistant that provides the user with information and control, privacy beacons that disseminate machine-readable privacy policies, and a privacy aware database (called

as *"citizen competence."*

PawDB) that stores and manages user data according to the agreed-upon collection and usage principles.

As one moves around in a ubiquitous computing environment, the *personal privacy assistant* will keep track of all data collections happening with and without the user's help. Data collections are announced either as part of the service protocol used (e.g., Jini, or as part of an RFID protocol), or through automated *privacy beacons* that continuously broadcast the corresponding privacy policies of a room or building via Infrared, Bluetooth, or Wireless LAN. Whenever possible, the assistant will enable or disable optional services, based on the user's preferences. Data solicitation and user control is provided through *privacy proxies* – continuously running services that can be contacted and queried by data subjects anytime, allowing them instant access to their data, e.g., to perform data updates and deletes, or query usage logs.

Once data has been solicited from the user (either actively by receiving a data submission via the privacy proxy, or implicitly by receiving sensor data such as video or audio feed), it is stored in PawDB, a back-end database that not only stores the data collected, but also each individual privacy policy that it was collected under. This allows the database to ensure that the promises made in a privacy policy with respect to the lifetime, usage, and recipient of a certain piece of information are kept, as well as provide users with a detailed "usage log" of their personal data (*recourse*).

As a proof-of-concept system, we have implemented our privacy awareness system PawS on a conventional PDA with Wireless LAN as the user assistant, infrared transmitters as privacy beacons, and a desktop computer for back-end computing (such as running the user and service proxies, as well as the PawDB). We have additionally verified our concepts in the domain of RFID-privacy, where we have augmented a standard reader-to-tag protocol to include privacy notices.

Outlook

What lies at the intersection of privacy protection and ubiquitous computing is easy to imagine: the frightening vision of an Orwellian nightmare-come-true, where countless "smart" devices with detailed sensing and far-reaching communication capabilities will observe an ever expanding part of our lives, so unobtrusive and invisible that we won't even notice. Ron Rivest calls this the "reversal of defaults": *"What was once*

private is now public", "what was once hard to copy, is now trivial to duplicate" and *"what was once easily forgotten, is now stored forever"* [293].

With our PawS prototype, an important component of a privacy supportive ubiquitous computing environment has been designed and implemented. A number of open questions remain, however, such as: How can we differentiate combined sensor readings from multiple people (e.g., a camera recording a meeting) that have chosen different sets of privacy policies? And how well can the average user specify and maintain her privacy preferences, especially in the context of invisible computers?

Despite such yet unresolved issues, PawS could already be useful as a silent but watchful transparency tool that keeps track of whom we leave our personal data with, thus allowing users to hold data collectors *accountable* to their privacy statements. Embedded in corresponding legal and social frameworks, such technical solutions can form the basic building block for a future with invisible computers that people can trust in.

The central tenet of every democracy in the end is trust.
Bill Clinton[3]

[3] As quoted in Gewn Ifill, "Bill & Al's Traveling Medicin Show," New York Times, 9 September 1993.

1 Introduction

The most profound technologies are those that disappear.
They weave themselves into the fabric of everyday life
until they are indistinguishable from it.
Mark Weiser[1]

The increasing miniaturization of computer technology will, in the foreseeable future, result in processors and tiny sensors being integrated into more and more everyday objects, leading to the disappearance of traditional PC input and output media such as keyboards, mice, and screens. Instead, we will communicate directly with our clothes, watches, pens, and furniture – and these objects will communicate with each other and with other people's objects. More than 10 years ago, Mark Weiser foresaw this development and described it in his influential article "Computer for the 21st Century" [351]. Weiser coined the term "ubiquitous computing," referring to omnipresent computers that serve people in their everyday lives at home and at work, functioning invisibly and unobtrusively in the background and freeing people to a large extent from tedious routine tasks. A more pragmatic approach is usually associated with the industry-initiated term "pervasive computing," which in principle follows the same goals as Weiser's ubiquitous computing, yet specifically tries to use existing or soon-to-be-available mobile-computing technologies. In its 1999 vision statement, the European Union's "Information Society Technologies Program Advisory Group" (ISTAG) used the term "ambient intelligence" in a similar fashion to describe a vision where "people will be surrounded by intelligent and intuitive interfaces embedded in everyday objects around us and an environment recognizing and responding to the presence of individuals in an invisible way" [15].

The vision of a future filled with smart and interacting everyday objects offers a whole range of fascinating possibilities. For example,

[1] In [351]

parents will no longer lose track of their children, even in the busiest of crowds, when location sensors and communications modules are sewn into their children's clothes. Similar devices attached to timetables and signposts could guide blind people in unknown environments by "talking" to them via a wireless headset [74]. Tiny communicating computers could also play a valuable role in protecting the environment, for example as sensors the size of dust particles that detect the dispersion of oil spills or forest fires [188]. Another interesting possibility is that of linking any sort of information to everyday objects, allowing for example future washing machines to query our dirty clothes for washing instructions [39]. While developments in information technology never had the explicit goal of changing society, but rather did so as a side effect, the visions of ubiquitous computing and ambient intelligence expressly proposes to transform society by fully computerizing it. It is therefore very likely that this will have long-term consequences for our everyday lives and ethical values that are much more far-reaching than the Internet with all its discussions about spam e-mails, cybercrime, and child pornography. With its orientation toward the public as well as the private, the personal as well as the commercial, the vision of an ambient-intelligence landscape aspires to create technology that will accompany us throughout our entire lives, day in and day out. And if Mark Weiser's vision of "invisible computing" actually materializes, we won't even notice any of it.

1.1 The Vision and Technology of Ubicomp

Mark Weiser's goal was to "make computing an integral, invisible part of the way people live their lives," because "only when things disappear in this way are we freed to use them without thinking and so to focus beyond them on new goals" [352]. Instead of carrying around specialized pieces of hardware with dedicated user interfaces (such as a keyboard and a screen), Weiser envisioned everyday items such as mirrors, garage door openers, and newspapers to become extended with new, helpful functionality: mirrors in my wardrobe display the local weather forecast while I am picking out my clothes for today, garage door openers can also locate the corresponding manual should the owner loose it somewhere in the house, and smart pens can tip on an article in the newspaper in order to receive an electronic version of it by email.

Computers as everyday artifacts would thus feature user interfaces

that were geared toward the actual use of the object, instead of using mechanisms that could equally well be used to write news articles (e.g., traditional keyboards) or draw lines (e.g., a mouse). Wireless communication technology would allow all these objects to interact with other computers on the Internet, as well as with other objects around them. Weiser also called this "embodied virtuality" in order to contrast it to the then much more popular field of virtual reality: "virtual reality is only a map, not a territory. It excludes desks, offices, other people not wearing goggles and body suits, weather, grass, trees, walks, chance encounters and in general the infinite richness of the universe. Virtual reality focuses an enormous apparatus on simulating the world rather than on invisibly enhancing the world that already exists" [352].

1.1.1 Technology Trends

While Weiser's "idea of integrating computers seamlessly into the world at large" was still relatively limited more than ten years ago, when Weiser introduced the concept of ubiquitous computing, today's technical progress makes many of his visions seem quite feasible.

One of the most important driving factors of this vision is probably the constant miniaturization in the field of microelectronics. Already in the later 1960s, Intel-founder Gordon Moore drew up what is known today as Moore's Law [246], predicting that the power of microprocessors would double every 18 months. So far, Moore's Law has not only held true with astonishing accuracy, but also expanded to apply to storage capacity and communications bandwidth as well. Conversely, prices for equivalent computing power or storage capacity have fallen radically over the last 40 years, allowing computing power to become a cheap, everyday commodity. While Weiser assumed that this trend might already subside in the mid-1990s [351], current predictions expect Moore's Law to continue to apply for at least another 15 years [234].

Equally important for realizing Weiser's vision of ubiquitous computing is the recent progress in material sciences, which will allow computers to come in novel forms that would not be associated with traditional computers anymore. Light-emitting polymers, for example, can be used to integrate displays in plastic foils, which can then be affixed to windshields, milk cartons, or cereal boxes [210]. Tiny charged, two-sided beads embedded in two layers of plastic can simulate the effect of pa-

per by being electrically oriented to display their black or white side toward the reader, without the need for continuous power supply [92]. Conductive fabrics allow designers to embed user interfaces directly into garments, such as buttons or thin foil displays, whose data would be routed through the garment without the need for additional cables.

Improved wireless communication technology such as wireless LAN, Bluetooth,[2] or the emerging ZigBee standard[3] allow for an increased interconnectivity between the various devices without wires or even central infrastructure components (ad-hoc networking). Radio frequency identification (RFID) tags can communicate with a corresponding RFID reader without having a battery of their own, simply by using the energy of the reader's electromagnetic field, thus allowing the remote reading of an item ID or tag-stored product information without a line of sight and without the need for an energy supply on the tagged object itself. Novel programming concepts like Jini[4] or UPnP[5] facilitate the discovery of and spontaneous interaction with previously unknown services, thus greatly increasing the capabilities of small devices.

Last not least, better and smaller sensors, some even without battery, greatly enhance the ability of small devices to perceive their environment and derive the context of their use. Capacitive fingerprint sensors allow for an unobtrusive user identification in a small form factor without having to enter a username and password.

Taking all this together, computers in the future could come in forms and functions that look very unlike today's desktop PCs, or even PDAs. At Brunel University, UK, students built a "smart toaster" that is connected to the Internet and puts today's weather forecast on each slice of bread [343]. Researchers at the Telecooperation Office at the University of Karlsruhe created a "smart coffee cup" that uses temperature sensors and ad-hoc wireless networking in order to determine whether a meeting (featuring many hot mugs in close proximity) is in progress [154] – something an equally smart mobile phone could use to switch to a "silent" profile [306].

Future technology will allow us to build artifacts and environments that can show context-aware behavior by sensing the user's actions and reasoning about his or her intentions, thus providing additional functionality appropriate to the current task. Artifacts and environments

[2] See `http://www.bluetooth.com/`
[3] See `http://www.zigbee.org/`
[4] See `http://www.jini.org/`
[5] See `http://www.upnp.org/`

will be able to "remember" events (e.g., encounters with humans or with other smart things) by using internal or external memory, and can "talk" about their experiences with other smart objects, smart environments, or with the user.

1.1.2 Societal Trends

Being able to build something in theory does not necessarily imply that it will become widespread in use. Future predictions on how technology will be used are notoriously imprecise, as popular science magazines of the 1950s and 1960s have demonstrated by predicting a future full of picture-phones, nuclear-powered cars, and even flying houses that would allow their inhabitants to migrate south in winter. At the 1964 world fair in New York, General Motor's "Futurama" pavilion described a future under the seas:

> "[There are] resort hotels, free-floating or secured to the ocean floor ... Through oversized windows vacationers may be seen dancing, eating dinner ... others are renting underwater camera equipment ... a swimmer is being towed by an aquascooter past a port at which a number of undersea vehicles are docked ... Within the lifetime of most Futurama visitors ... man will make his greatest advances on Earth itself ... the seas will be harvested and mined" [304].

While a future envisioned by proponents of ubiquitous computing computing might seem equally outlandish to some, a number of important societal trends exist that might make adoption of this new technology much more likely than past technology adoption predictions have proven to be:

- *Efficiency.* Smart environments and smart artifacts can potentially improve the efficiency in the areas of supply chain and inventory management [120]. Tracking crates, packages, and individual items with RFID tags allows companies to monitor their assets in real-time, not only alleviating the need for costly inventory taking, but also facilitating intra- and inter-company data exchange in order to smooth out demand fluctuations in the supply chain.
- *Convenience.* Efficiency of a different sort can be provided by PDAs and mobile phones, allowing us to better manage both our

professional and private lives. Smart phones that allow us to better stay in touch with our loved ones, while also providing our employer with, say, information about our current location during office hours so that we are not required to be at our desks all the time, can be an important sales argument for smart devices.

- *Security.* Advances in telecommunications, Internet commerce, and air travel have resulted in our lives being more and more dependent on interactions with previously unknown people. Smart environments and smart artifacts can not only help us assessing the trustworthiness of others, but also tirelessly monitor the conditions of goods, containers and buildings, thus helping both law enforcement and emergency personnel.

While the success or failure of ubiquitous computing applications – as with any prediction of the future – is far from clear, these attributes of efficiency, convenience, and security play an increasingly important role in today's society and might thus provide an important driving factor for the increased proliferation of ubiquitous computing technologies in the near term future.

1.2 Social Issues of Smart Environments[6]

A future where our everyday lives are comprehensively digitized might have significant social consequences. Beyond the obvious implications for our privacy (due to the large amount of data collected by smart artifacts, see section 2.2), our health (due to the effects of increased non-ionized radiation, see [165]), and our environment (due to the accumulation of heavy metals in our homes and offices, and ultimately in our waste facilities [165]), a range of additional issues play an important part when it comes to judging the social impact of ubiquitous computing technologies. While not part of this thesis, we nevertheless want to give a broader perspective of the issues involved.

1.2.1 Reliability

Rather than a single device that contains a large number of features, ubiquitous computing systems are typically comprised of a significant

[6] This section is based on joint work with Jürgen Bohn, Vlad Coroamă, Friedemann Mattern, and Michael Rohs [40].

number of (functionally) smaller systems that can instead cooperate to achieve a similar, if not superior, functionality than the monolithic system. Additional advantages are the mix-and-match approach of being able to easily combine different devices for a given task, and the higher redundancy of cheap, interchangeable components that should allow users to easily substitute a malfunctioning or lost device, e.g., with a similar one from a friend.

However, as the number of different system components from different manufacturers, owners, and technology platforms increase, the chances for system failure increases as well. Specifically, Bohn et al. [40] cite four reliability issues of future ubiquitous computing systems:

- *Manageability:* As the number of possible interactions between ubiquitous computing devices increases quadratically, it is far from clear whether we will still be able to manage the individual functions and interconnections. Also related to this point is the issue of dynamic functionality, where devices do not have a fixed use per se but can be instrumented in a large number of situations by combining them with other tools. This will both challenge users and programmers alike.

- *Predictability:* An important factor for human tool usage is the issue of predicting its actions, both for the successful application of a tool, as well as for the case of failure. Knowing how a certain device will work, especially in combination with other devices or the environment, can be a crucial factor for future ubiquitous computing systems. This will be especially important in case of a system failure: Not being aware of a malfunctioning or deactivated system might entail serious consequences for users relying on its operation. However, given the unobtrusive nature of such systems, a direct feedback loop such as a telephone dial tone (which indicates that both the phone and the infrastructure are working) might be difficult to realize in such scenarios.

- *Dependability:* The functioning of a smart device is further complicated by its often severe resource restrictions. Small form factors and low power budgets leave not much room for safety margins. In addition, the use of mobile devices in non-office environments will increase the chance of a physical device failure (e.g., a dropped devices or water damage). Bohn et al. call for a higher factor of

redundancy when designing such systems in order to overcome the expected larger number of device failures.

- *Autonomy:* As artifacts rely more and more on the presence of other devices or being in reach of a supporting environment in order to function, their increased dependence on outside services would reduce the "object constancy" for the user, as the functioning of a smart device would rely on a large number of secondary factors. For example, an electronic book might require connectivity to a license server in order to display a certain page, and might therefore occasionally fail to work.

1.2.2 Control

Ubiquitous computing technology promises to ease the burden of daily chores to the user by automating routine tasks and letting humans focus on the problems they are trying to solve, not on the actual process of solving it. Rich Gold, artist in residence at Xerox PARC, put this ambition quite succinctly:

> "Nobody wants a drill. They want the hole that it produces. Nobody really wants a printer. They are big and expensive. But they do want documents. They need the printer to produce the documents. Actually, there are researchers at PARC who would claim that people don't even want documents. They want the knowledge that documents contain or create."[7]

However, this explicit goal of hiding the actual operation of ubiquitous computing systems from the user creates a number of problems in turn. For one, it complicates the above reliability issue as it counteracts any attempt at direct tool manipulation (and thus, a closely-coupled feedback cycle). But Bohn et al. find three additional issues with this approach that stem more from social engineering issues than from technical difficulties:

- *Content Control:* With smart artifacts and rooms providing a wealth of information to other smart devices, environments, and ultimately to the user, they become a medium just like newspaper, radio, or television. The unobtrusive nature of interactions

[7] See `http://www2.parc.com/red/`

will make it difficult to judge the extend of influence any such information will have in our lives. Additionally, the complex system interactions might substantially complicate tracing the information flows to its origin, thus preventing consumers to judge the reliability of thusly acquired information.

- *System Control:* Just like today's digital rights management systems control the way we handle our digital documents, digital music, and digital video, future smart artifacts might control not only our digital assets, but also our real-world belongings. A smart car might control our speeding behavior by limiting its maximum velocity to the current speed limit, while a cheap smart fridge might restrict the brand of milk it can order to the company that provided the fridge as part of a sales event.

- *Accountability:* The autonomy of smart devices will make it difficult to detect the actual locus of control when it comes to issues of accountability. This obviously relates to malfunctions of the system (a smart doll ordering large numbers of accessories on the Internet, or a smart fridge buying excessive amounts of food online), but also applies to the "intended" use of the system in case manufacturer, service provider, and owner disagree with respect to what constitutes a "normal" operation. The fact that potentially a large number of devices and services can be involved in a single transaction further complicates the search for someone to hold accountable.

1.2.3 Social Integration

Future computing promises to be inclusive rather than exclusive: instead of having to learn how to use specialized equipment such as keyboards, mice, and windowed operating systems, smart items are envisioned to adapt themselves to their human users, so that interacting with their embodied artifacts becomes a natural task for their owner without the need to specifically adjust to their special capabilities.

In practice, however, a comprehensive digitization of our everyday lives, even if it explicitly tries to ease the utilization of such computerized environments and artifacts, will pose significant social challenges that go beyond simple ease-of-use. [40] lists a number of such integration problems:

- *Transparency:* Even trivial transactions might become incomprehensible, simply through their scale of minuscule interactions. With potentially hundreds of miniature transactions taking place at any moment, the ability to inspect and verify their correctness significantly decreases. This is similar to the loss of system control described in the previous section, though instead of applying to the momentary operation of the system, it focuses on the post-hoc validation.

- *Knowledge Sustainability:* The more dynamic our environment gets, the less useful past experiences will become. Without being able to sustain our knowledge, long-term experiences might become devalued, thus increasing uncertainty and disorientation in our everyday lives. This development might accelerate our dependency on smart environments that help us cope with these new dynamics.

- *Fairness:* The trend for personalization through customized environments and attentive smart devices not only brings the benefits of sparing us from uninteresting options and services, but also has the potential of reinforcing inequality by withholding from us information that we are not "worthy" enough receiving. David Lyon, Professor of Sociology at Queen's University in Canada, calls this process "social sorting:" "Categorizing persons and groups in ways that appear to be accurate and scientific, but which in many ways accentuate differences and reinforce existing inequalities" [228].

- *Universal Access:* While easier access to information should narrow the "digital divide,"[8] the increased complexity of devices, systems, and data flows in the future might actually increase the burden for the elderly and less technically inclined, when not using such systems ceases to be an option. Similarly, it could become increasingly difficult to assess the trustworthiness of information that one obtains through smart environments, especially for lower income households that might need to rely on advertisement-supported services. Due to the increased digitization of the real world, today's digital divide might thus increasingly create a rift in our real-world society.

[8]The term "digital divide" refers to socio-economic inequalities between people who have access to computers and those who do not [365].

- *Man-World Relationship:* Philosophers also criticize the tight coupling of ourselves with the environment, seeing the extensive instrumentalization of the world around us that ubiquitous computing seeks as something that will ultimately lead to "a transformation, dislocation, substitution, and the loss of fundamental properties relating to the world" [6].

- *Rebound Effect:* Many technologies that once were thought to free us from laborious tasks, or save us time by accelerating a time consuming process, have turned out to actually increase the burden of our daily chores. Hilty et al. [165] call this the "rebound effect:" an increased efficiency resulting in a cheaper product subsequently leads to an increased demand, thus canceling out or even surpassing the envisioned savings in time or raw material. Examples are more efficient traffic guidance systems that lead to increased road traffic (thus congesting roads even more), or electronic mail that minimizes time and cost of communication but leads to a large increase in email-traffic that surpasses any amount of time formerly spent on writing letters.

This shows that beyond the maybe obvious privacy implications, a future full of ubiquitous computing systems will most likely also significantly affect the fabric of our everyday, through issues of reliability, control, and social integration. While not directly part of this work, these arguments will nevertheless have to be kept in mind throughout the rest of our discussion. In the last section of this chapter, we want to take a look at how designers and developers of ubiquitous computing systems deal with this multitude of social issues, and in particular privacy issues, both from a conceptual point of view as well as from a practical perspective.

1.3 Designing Privacy-Aware Systems

Even though few projects in ubiquitous computing explicitly address privacy in their research agenda, many designers of such system openly acknowledge the implications for privacy and reiterate their concern for these issues. However, this concern has so far resulted in few efforts to build privacy-aware smart environments and smart devices.

As part of this dissertation, the author visited four different European research projects in the field of ubiquitous computing, in order to

interview researchers in the field regarding their views on privacy, responsibility, and potential solutions [207]. Though it was initially envisioned that the visits would result in an explicit account of the "state of privacy" of the cumulative minds of researchers in ubiquitous computing, few researchers had actually thought about such problems enough to be able to provide additional insights. This lack of responsibility for such issues provides an important driver for this work: As developers of ubiquitous computing systems do not have privacy on their agenda, ready-made solutions and guidelines will play an important role when fielding ubiquitous computing prototypes in the future.

1.3.1 Taking Stock: The DC-Troubadour Action

In January 2001, the European Union launched the ambitious "Disappearing Computer Initiative," a three-year program to fund 17 projects in the area of ubiquitous computing *"to see how information technology can be diffused into everyday objects and settings, and to see how this can lead to new ways of supporting and enhancing people's lives that go above and beyond what is possible with the computer today."*[9]

At the first Disappearing Computer (DC) conference in Zurich in October 2001, a group of researchers organized a privacy workshop as part of the meeting program, with the goal of surveying privacy implications of the various DC projects. However, it turned out that an individual project assessment was nearly impossible without knowing the exact details and provisions of its systems and prototypes. Thus, the idea of a privacy troubadour was born: having a dedicated researcher visit individual DC-projects, it should be possible to answer in detail questions like "Where is data stored?" "Who has access to this data?" or "How long is data retained?", which all seemed to be required to judge a project's privacy implications. Beyond such factual project data, the group moreover hoped to be able to harness specific design experiences with respect to privacy: whenever a decision to process or store personal information or sensory data was made as part of the system design, the people involved would probably have made some technical or moral judgment as to its effect on user privacy. The group members envisioned soliciting such implicit concerns and unspoken ideas to arrive at privacy guidelines that would have been created from practical experience instead of theoretical analyses.

[9] See `www.disappearing-computer.net`

October 2002	Ambient Agoras, Paris, France (internal meeting)
November 2002	Smart-Its, Lancaster, UK
December 2002	Oresteia, London, UK
January 2003	Smart-Its, Gothenburg, Sweden
February 2003	E-Gadgets, Patras, Greece
May 2003	Ambient Agoras, Paris, France (internal meeting)
May 2003	Interliving, Paris, France

Table 1.1: *Privacy Troubadour Visiting Schedule.* During October 2002 and March 2003, five different DC projects at six different locations were visited, in order to learn more about DC designer's approach to privacy.

The Privacy Troubadour Action (TA6) within the Disappearing Computer Initiative was granted in September 2002. The initial application document proposed that

> "...by visiting selected projects within the DC-community, the troubadour should be able to examine each project's individual goals and concepts in detail in order to establish its inherent privacy threats and suggest improvements. Visits would include demonstrations of existing prototypes and various discussions with developers and researchers concerning their project goals and implementation methods."[10]

Its initial funding included visits to five different DC projects in a first round effort, with an optional extension of visiting another five projects after a mid-term report had been prepared. Initial contacts were made during the second DC jamboree, held in October 2002 as part of the Ubicomp 2002 conference in Gothenburg. The interest in such an activity was actually quite high, and many projects welcomed a visit from the troubadour as they all were concerned about privacy implications and were eager to learn more about the issue, as well as share their design experiences.

Table 1.1 gives a timeline of the first seven visits, including two preparatory, internal meetings. The format of the visits varied widely. At one site, a single researcher was available for most of the day to exclusively discuss a previously sent questionnaire with the troubadour, while other sites had arranged for a large number of meetings with different researches, also from non-DC-projects. As part of each visit, interviews were recorded for later transcription, totaling about five hours of audio.

[10]See `www.inf.ethz.ch/~langhein/projects/dc.html`

	Not possible	**Not necessary (yet)**	**Not necessary (at all)**
Ethical reasons	(Perfect) privacy is an illusions, so no obligation to create	No issue for research prototypes, as they don't work in the real world with real data	Not a technical issue: this must be solved by society via legal and moral guidelines
Technical reasons	Too complex of a problem to be solved technically	Privacy issues will only become relevant when initial issues (networking, energy, etc.) are solved	Solutions exist in other areas (e.g., Internet technology) that can be used in Ubicomp
Resource constraints	No funding for privacy issues	Technical issues more pressing at the moment	Not part of the project deliverables

Table 1.2: *Responsibilities and Excuses – Hypotheses Grid.* Designers of ubiquitous computing systems have various reasons for not working on privacy related issues in their projects.

1.3.2 Responsibilities and Excuses

The initial aim of learning about the individual experiences of DC-researchers in order to arrive at privacy guidelines for future DC-projects soon proved futile. Most researchers that participated in the interviews and discussions did not (yet) think of privacy issues in their own work, or only on a very obvious level. Over the course of the various interviews and discussion, the following hypotheses emerged that would explain why researchers, even with a heightened awareness for privacy issues, would not actively pursue the privacy implications of their systems (summarized in table 1.2):

- *Not feeling morally responsible:* There were several reasons why researchers felt that it was not up to them to provide for privacy awareness in their designs: either lack of applicability to their specific field of expertise ("for [may colleague] it is more appropriate to think about privacy issues. it is not really the case in my case") or because other social processes were felt to be more adequate to regulate such issues ("little by little – I expect that would be a process of 20 years – that you need a generation actually to sort out, where is the social value, [...] and then formalize the legislation").
- *Not necessary anymore:* Some researchers thought existing security mechanisms to be adequate protection from privacy abuses: "I think all you need is really good firewalls. [...] if you know, or

if you are aware of, that this might be a problem, then you are safe." Similar ideas came up in other interviews: Question: "So you imagine that existing technology would be used?" Answer: "Yes, right."

- *Not yet necessary:* In many cases, researchers thought that only after initial prototypes had been built, a topic like privacy could properly be addressed. One of the many design strategies heard were: "we first thought: let's build this first..." and "my approach is more to really build these things now in order to see what issues arise there."

- *No problem for prototypes:* Related to the above point, but with a slightly more practical orientation, were remarks that privacy had not proved to be a problem in this early stage of prototype design. Far more often, designers would identify and tackle problems of energy usage, communication protocols, or data analysis, instead of spending creative energy on privacy issues.

- *Too abstract of a problem:* In some cases, researchers purposefully did not think about privacy: "I think you can't think of privacy when you are trying out... it's impossible, because if I do it, I have troubles with finding ubicomp future [laughs], when I think of the privacy issues. but i... and the more I think about it, the more I become skeptical. but... on the other hand, some.... I think it's important that you think about it, but I think you can't... you can't... when you are building prototypes and you are trying making design examples you can't have that..."

- *Not part of deliverables:* In one case, four hours had been reserved for privacy issues during a two-day meeting. However, the first day the session got cut down to half the time due to extended discussions on getting the final deliverable into shape. The second day saw the entire rest of the planned privacy session canceled, due to ongoing deliberations about specific implementation details. In another case, interviews were cut short since the researchers had to furnish the newly acquired office space (e.g., unpacking boxes, rushing to IKEA to buy new furniture...).

The few cases that had researchers explicitly address privacy were few and often shallow. Some projects had privacy listed as part of one

of the deliverables, so a general note on privacy definitions and issues, as well as a brief description of ethics, had been produced. However, during the continued development of the prototype, no re-evaluation of the system in light of these issues, or a re-evaluation of such issues in light of the existing prototype was made.

Apart from the aim of gathering implicit knowledge from researchers, the idea of directly asking specific implementation details in order to evaluate a projects privacy invasiveness also turned out to be rather ineffective. In most cases, design choices pertaining to privacy, like data storage and dissemination, were not fully specified yet. Even though prototypes might be storing or communication personal sensory data in a specific way, most designers pointed out the temporary nature of such arrangements, which would of course be redesigned should their prototype ever be used in a production system.

1.3.3 Conclusions and Concerns

The troubadour grant application stated that *"a troubadour is not sent as a lecturer offering ready-made solutions to existing privacy threats within a project, but instead be a collaborator of the regular project members trying to increase the social acceptance of the project"* [207]. While the reception at all projects was warm and quite often with genuine interest in the topic, the lack of privacy *requirements* in most projects turned out to short-circuit the idea of *collaboratively* sorting out the problem of privacy in the Disappearing Computer initiative: input from the troubadour was welcome, but few people had the time and energy to substantially analyze their own work.

As long as privacy is situated on a non-critical development path, more important issues such as energy efficiency, code size, or robustness dominate the researcher's todo-lists. Decisions pertaining to data storage and communication details are often improvised and seen as a temporary solution fit for prototype deployment. Projects which explicitly had privacy issues as part of their deliverables, generally exhibited greater concern for such issues, even though they often stopped short of generating novel ideas and limited themselves to a broad but shallow summary of general privacy issues, without taking project specific design parameters into account.

If a robust culture of privacy awareness is to be fostered among designers of ubiquitous computing systems, making such requirements

explicit already as part of the project funding process seems to be the most viable approach. Even if designers feel morally responsible, unless either users (in a comprehensive field study) or project officers ask for it, there will hardly be much time and energy to spare. Having a better set of requirements to test prototype systems against would also contribute to the cause, though such technical issues would probably better be tested by a thorough examination of project documentation, together with singular interviews for clearing up specific implementation details.

While many designers claim that *"these are only prototypes,"* with no connection to real users and thus no "real" privacy concerns, the lack of awareness for privacy issues at this stage nevertheless prompts several concerns:

1. *Bad publicity:* As a number of recently fielded prototypes in the area of RFID-based tracking have shown, having no proper privacy protection mechanisms in place can result in a serious public backlash [42, 124, 382], ultimately tarnishing an organization's image in the public eye.

2. *Public lack of trust:* Trust in organizations and institutions is an important factor for any democratic society, not only in terms of political stability [275], but also for its positive influence on economic performance [127], low crime rates [297], or public health [193].

3. *Lack of privacy culture:* Many research prototypes eventually find their way into a product, usually going through various stages where researchers gradually hand over development to product engineers and marketers. Building privacy principles already into the initial design can positively influence the entire product development, rather than having privacy issues completely ignored or later added due to external pressures.

By thoroughly discussing the long history and different facets of privacy, this thesis also hopes to contribute to an improved understanding and heightened awareness among researchers in the field of ubiquitous computing, such as the ones interviewed during the DC troubadour activity.

1.4 Summary

Ubiquitous computing technology has the potential to significantly alter our everyday. It could change the way we think about, perceive, and interact with computers by letting them take new forms, provide them with invisible communication capabilities and allow them to sense and reason about the environment they are deployed in.

But it is only a possibility, just one of many ways in which our future might develop. The potential is there, but so it seemed thirty years ago, when we felt on the verge of moon colonization, undersea cities, and nuclear powered cars. But this time there are good reasons to pursue this vision. It promises more efficient development, productions, and sales. It seems to provide help to those who are looking for a more convenient life. And it is already in use to increase the safety and security of airports, buildings, cars, and homes (section 1.1).

Yet with all its potential to do good, its way of integrating the digital world with our everyday has its risks, though many of them are only dimly recognizable yet, at the very outset of its deployment (section 1.2). Knowing the reliability of today's PCs, with their frequent hardware and software failures, and their vulnerabilities to outside attacks and insider break-ins, would such a vast computerization of our lives ever work reliably? How could we control and manage that many computer systems? And how would this change our daily interactions and experiences that we have come to rely upon over hundreds of years, ways that change much slower than the pace of technical evolution?

Privacy is but one concern of this miasma of social issues, yet maybe one of the more pressing. It lies at the very core of any ubiquitous computing future, as the comprehensive digitalization of our everyday actions forms the basis for almost any application in its domain. And few of those in position to recognize this threat and do something about it seem yet concerned. Lawyers and politicians still struggle to make sense of the new borderless realities the commercialization of the Internet has created, while technologists mostly ignore the issue as of now (section 1.3): It gets in the way of research as it limits potential developments; it is too abstract of a problem to take into account yet; or it is a question of policy, not technology, to worry about.

This chapter has tried to prepare the setting for our thesis: That ubiquitous computing will significantly affect our future privacy, and that we have to actively pursue the integration of privacy-awareness

into our ubiquitous computing infrastructures. In the following pages, we want to outline how such an integration might be realized. An integration that would allow us to change little in the way we handle our personal information in order to control who knows what about us. An integration that might even make it *easier* for us to keep track of our personal data flows. An integration that builds upon the interplay between social norms, legal protection, and technology to achieve what it sets out to do. Or according to security expert Jim Morris of Carnegie-Mellon University:

> "[To] build computer systems to have the same privacy safeguards as the real world, but no more, so that ethical conventions will apply regardless of setting. In the physical world, for example, burglars can break through a locked door, but they leave evidence in doing so" (as cited in [351]).

1.5 Thesis Outline

After having presented our case in chapter 1, the following chapter will begin our in-depth analysis of the problem: What are the roots of modern privacy, its history and definitions, and how is privacy protection implemented in today's society? It will also describe in detail how both the vision and the technologies of ubiquitous computing will affect our privacy once such systems get deployed.

Chapter 3 then looks in detail at the various mechanisms available to us in each of the domains that span the solution space: Social principles to control privacy, legal protection to punish violators, and technical tools that support these mechanisms. Based on these findings, the chapter builds up a set of guiding principles in which we want to embed our technical privacy-supportive infrastructure, establishing it as an integral part of a comprehensive suite of protection tools in the technical, social, and legal realm.

Chapter 4 finally presents our technical architecture to support today's social and legal norms for protecting privacy. It restates our requirements, draws up the general architecture, and provides a detailed look at our prototype implementation before discussing its benefits.

In chapter 5, we then look at alternative approaches to privacy for ubiquitous computing systems, as well as at related work from the fields of user interfaces, computational trust, and databases, and compare it

with our work on PawS. We will also discuss in more detail the issues of location privacy and RFID privacy, as well as describe the currently proposed solutions.

Chapter 6 will then outline how our core principles in PawS can be applied in the area of RFID privacy, and contrast our approach to the solutions presented in chapter 5.

A summary in chapter 7 closes our argument, reiterating the most pressing issues in the context of privacy protection in ubiquitous computing, restating our principles, and enumerating future work.

Sections of this thesis have been published in the following workshops, conferences, and journals:[11]

- *Motivation*: When Trust Does Not Compute – The Role of Trust in Ubiquitous Computing [207]
- *Analysis and implications*: Digitalisierung des Alltags. Was ist Pervasive Computing? (joint work with Friedemann Mattern) [210], Living in a World of Smart Everyday Objects – Social, Economic, and Ethical Implications (joint work with Jürgen Bohn, Vlad Coroamă, Friedemann Mattern, and Michael Rohs) [39]
- *Privacy principles*: Privacy by Design – Principles of Privacy-Aware Ubiquitous Systems [205]
- *Trust*: When Trust Does Not Compute – The Role of Trust in Ubiquitous Computing [208]
- *RFID-Privacy*: Die Privatsphäre im Ubiquitous Computing – Datenschutzaspekte der RFID-Technologie [209], Scanning with a Purpose – Supporting the Fair Information Principles in RFID Protocols (joint work with Christian Flörkemeier and Roland Schneider) [123]
- *Privacy-Awareness System*: A Privacy Awareness System for Ubiquitous Computing Environments [206]

[11] Some articles have originally been published in German.

2 Background and Analysis

When I use a word, it means just what I choose it to mean – neither more nor less.
Humpty Dumpty[1]

Protecting people's privacy is a very personal affair. Something that cannot be solved without taking people's habits, preferences, and moral views into account. One approach would be to conduct in-depth interviews about individual preferences and perceptions regarding privacy. Especially in the context of Internet privacy, a wealth of such polls exist [84, 156, 333]. However, as many of these surveys have shown, people's unconscious handling of privacy often differs significantly from their conscious replies to direct questionnaires.[2]

An alternative approach is thus to look at privacy perception on the macro level – how society has handled the balance between the public and the private over the past hundred years and more, how its thinkers have tried to define it, and how they set out to motivate it. We will see that the protection of personal privacy is by no means just a recent trend of the information society, as debates over privacy have had a long history, during which changes in society and also technology have repeatedly altered society's conception of the reach and limits of privacy.

Most importantly, we hope that by approaching the problem of privacy first from a *conceptual*, rather than from a technical perspective, we will be better able to root our (technical) solution in social habits, norms, and routines. Developing a "social" technology such as our pri-

[1] In Lewis Carroll's "Alice Adventures in Wonderland." Quoted in [238] in order to point out that the terms "privacy" and "private" are used so loosely in everyday conversation that anyone who use them may claim, just like Humpty Dumpty does, that they mean whatever the person wants them to mean.

[2] A prime example for this is the high popularity of so-called club cards or loyalty cards, which provide shoppers with often less than one percent of discount in exchange for their detailed shopping records. According to recent data, more than 52% of German shoppers own at least one card [334], despite claims that more than 87% of German citizens value their privacy highly [156].

vacy solution without analyzing the social structures it is embedded in – its history, its motivations, and its daily realities – might run the risk of providing only a shallow and short-lived remedy.

We want therefore to begin our analysis of privacy in ubiquitous computing with a review of privacy in the literature, both its history and definitions, in order to motivate its place even in a future that might be very different from today. The core of this chapter then tries to analyze the impact of ubiquitous computing technologies on these definitions and social realities. After reviewing the range of existing privacy mechanisms in chapter 3, this analysis will form the basis on which we will build our set of guidelines that will steer our system development in chapter 4.

2.1 Privacy History and Definitions

While references to privacy can already be found in the Bible,[3] the earliest reference in common law[4] can be traced back to the English *Justices of the Peace Act* of 1361, which provided for the arrest of eavesdroppers and peeping toms [212]. In 1763, William Pitt the Elder, at that time member of the English parliament, framed in his speech on the Excise Bill the privacy of one's home as follows:

> The poorest man may in his cottage bid defiance to all the forces of the Crown. It may be frail — its roof may shake — the wind may blow through it — the storm may enter — the rain may enter — but the King of England cannot enter! — all his forces dare not cross the threshold of the ruined tenement [49].

This century-old tradition of respecting the privacy of the individual is by now enshrined in many local habits, national laws and international treaties, which have been put into place in order to fulfill this basic human need. The most prominent of these might be the Universal Declaration of Human Rights, adopted by the United Nations in 1948, which states in its Article 12 that:

> No one shall be subjected to arbitrary interference with his privacy, family, home or correspondence, nor to attacks

[3] See for example Revelations 2(17), Ephesians 5(31–32), Proverbs 20(19), and Proverbs 25(9).

[4] The *common law* is the legal system of many anglo-american countries. It is based on traditions and customs, dating back to historic England [364], and heavily relies on precedents.

> upon his honour and reputation. Everyone has the right to the protection of the law against such interference or attacks [335].

Postwar Europe saw privacy firmly established as a critical human right through Article 8 of the Council of Europe Convention of 1950 [75] and again in 2000 with the European Union Charter of Fundamental Rights,[5] which for the first time in the European Union's history sets out in a single text the whole range of civil, political, economic, and social rights of European citizens and all persons living in the European Union [322]. Article 8 of the Charter, concerning the Protection of Personal Data, states [104]:

1. Everyone has the right to the protection of personal data concerning him or her.

2. Such data must be processed fairly for specified purposes and on the basis of the consent of the person concerned or some other legitimate basis laid down by law. Everyone has the right of access to data which has been collected concerning him or her, and the right to have it rectified.

3. Compliance with these rules shall be subject to control by an independent authority.

Contrary to its recent prominence in Internet-related news, privacy is no short-lived, exaggerated side-effect of Web commercialization, even though it has certainly gained in relevance and public awareness through this development. However, its rich history and sometimes complex interdependence with the social fabric of our everyday lives is often reduced to the availability of sufficient security mechanisms, especially in the technical literature. Implementing authentication and encryption mechanisms is many times seen as the straightforward solution to an, after all, simple problem. Yet even though security is an important part of any privacy aware technical infrastructure, limiting privacy only to security does not provide an adequate translation of the many different facts of modern privacy. In front of the backdrop of its rich history, the next sections will try to provide a more nuanced view of this often elusive topic.

[5] Available at `www.europarl.eu.int/charter/`

2.1.1 Facets of Privacy

One of the earliest definitions of privacy comes from the later U.S.-Supreme Court judge Louis Brandeis, and his colleague Samuel Warren. The two published in 1890 the essay "The Right to Privacy" [345], which created the basis for privacy tort law in the U.S. legal system.

The fact that their essay is still very relevant today also stems from the circumstances under which Warren and Brandeis felt compelled to write it:

> Recent inventions and business methods call attention to the next step which must be taken for the protection of the person, and for securing to the individual what Judge Cooley calls the right 'to be let alone.' ... Numerous mechanical devices threaten to make good the prediction that 'what is whispered in the closet shall be proclaimed from the house-tops' [345].

What may sound like an accurate description of the new possibilities of ubiquitous computing systems, is actually a reference to the technical progress in the field of photography at that time. Before 1890, getting one's picture taken usually required visiting a photographer in his studio and sitting still for a considerable amount of time, otherwise the picture would be blurred. But on October 18, 1884, George Eastmann, the founder of the Eastman Kodak Company, received U.S.-Patent #306 594 for his invention of the modern photographic film. Instead of having to use the heavy glass plates in the studio, everybody could now take Kodak's "Snap Camera" out on the streets and take a snapshot of just about anybody without their consent. It was this rise of unsolicited pictures, which more and more often found their way into the pages of the (at the same time expanding) tabloid newspapers, that prompted Warren and Brandeis to paint this dark picture of a world without privacy.

Today's developments of Smart Labels, Memory Amplifiers, and Smart Dust seem to mirror the sudden technology shifts experienced by Warren and Brandeis, opening up new forms of social interactions that change the way we experienced our privacy in the past. However, even the strong resemblance of technological progress cannot ignore the fact that their "right to be let alone" looks hardly practicable today: With the multitude of interactions in today's world, we find ourselves constantly in need of dealing with people that do not know us in per-

son, hence require some form of information from us in order to judge whether such an interaction would be beneficial. From opening bank accounts, applying for credit, obtaining a personal yearly train pass, or buying books on-line – we constantly have to disclose part of our personal information in order to participate in today's life. Preserving our privacy through isolation is just not as much an option anymore as it was one hundred years ago.

Procedural Facets

A more up-to-date definition thus comes from the 1960s, when automated data processing first took place on a national scale. Alan Westin, professor emeritus of public law and government at Columbia University, defined Privacy in his groundbreaking book *Privacy and Freedom* as follows:

> Privacy is the claim of individuals, groups, or institutions to determine for themselves when, how, and to what extend information about them is communicated to others [354].

This definition is often described as *information privacy*, contrasting it to Warren and Brandeis definition of privacy as solitude, of being "let alone." While solitude might be an effect of information privacy, Westin stressed the fact that *"the individual's desire for privacy is never absolute, since participation in society is an equally powerful desire"* [354].

However, as Warren and Brandeis' definition suggests, being in control of one's personal data is only one facet or privacy. Back in the 19th century, the protection of the home – or *territorial privacy* – was the most prevalent aspect of privacy protection. Equally important was the idea of *bodily privacy*, the protection from unjustified strip searches or medical tests or experiments (e.g., drug testing). These two facets are also often called local, or physical, privacy. And with the invention of the telegraph and telephone in the late 19th century, the rise of modern telecommunication required re-evaluation of the well-known concept of *communication privacy*, previously manifested in the secrecy of sealed letters.

Functional Facets

Another way of differentiating the various conceptions of privacy can be found by distinguishing the various effects privacy has on our lives.

The above procedural facets are grouped around the three functional concepts of *zonal*, *relational*, and *decisional privacy*.

Zonal privacy protects certain spaces, such as our home, our workplace, or our car. Relational privacy protects the relationships in an individual's life, such as intimate family relations between husband and wife, or between mother and child. Decisional privacy is what Beate Rössler, professor for philosophy at the University of Amsterdam, calls *"securing the interpretational powers over one's life,"* the freedom to decide for oneself *"who do I want to live with; which job to take; but also: what clothes do I want to wear"* [298].

Privacy is thus also about the autonomy of the individual, about protecting our independence in making choices central to personhood. Westin describes this as follows:

> Each person is aware of the gap between what he wants to be and what he actually is, between what the world sees of him and what he knows to be his much more complex reality. In addition, there are aspects of himself that the individual does not fully understand but is slowly exploring and shaping as he develops [354].

This also connects with what Westin calls the *emotional release* functionality of privacy, moments "off stage" where an individual can be himself, finding relief from the various roles he plays on any given day: *"stern father, loving husband, car-pool comedian, skilled lathe operator, unions steward, water-cooler flirt, and American Legion committee chairman."* Equally important in this respect is the "safety-value" function of privacy, e.g., the *"minor non-compliance with social norms"* and to *"give vent to their anger at 'the system,' 'city hall,' 'the boss':"*

> The firm expectation of having privacy for permissible deviations is a distinguishing characteristic of life in a free society [354].

Constituent Facets

Another way of describing privacy is through its individual constituents. Ruth Gavinson defines privacy as being comprised of solitude, anonymity, and control [132]. Arnold Simmel puts it similarly, yet expands somewhat on Gavinson:

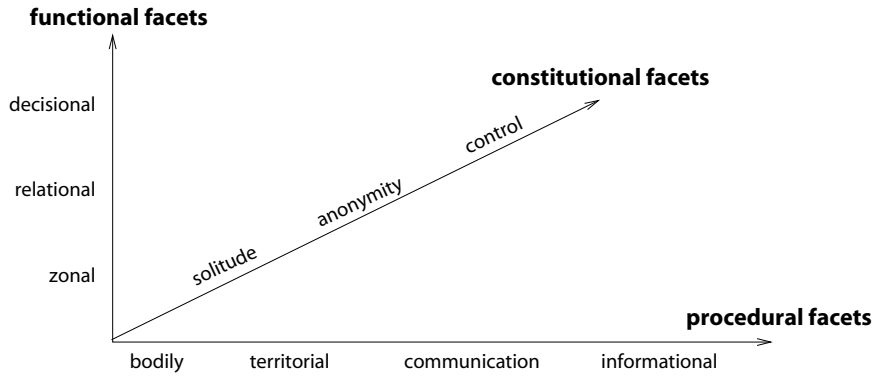


Figure 2.1: *Privacy Facets.* The privacy space can be subdivided along procedural, functional, and constituent facets. Note that facets along an axis can also overlap, which cannot be properly represented in the chosen three dimensional representation.

> Privacy is a concept related to solitude, secrecy, and autonomy, but it is not synonymous with these terms; for beyond the purely descriptive aspects of privacy as isolation from the company, the curiosity, and the influence of others, privacy implies a normative element: the right to exclusive control to access to private realms [319].

Contrary to Westin and Rössler, Gavinson and Simmel describe privacy not as an independent notion, but rather as an amalgam of a number of well established concepts, something that constitutes itself only through a combination of a range of factors. While Westin also relates privacy to concepts such as solitude, group seclusion, anonymity, and reserve [57], he calls them *privacy states*, indicating that these are merely different sides to the same coin.

Perhaps a synthesis of constitutional and functional description comes from David Flaherty, the data protection commissioner for British Columbia. Looking for information-related privacy interest reflected in the literature, Flaherty lists no less than thirteen privacy aspects, shown in table 2.1 [119].

2.1.2 Data Flows and Their Borders

Instead of looking at definitions of privacy, we can also try to grasp its meaning by examining when people feel that their privacy has been violated. Just as security, privacy is not a goal in itself, not a service that people want to subscribe to, but rather an expectation of being in

The right to individual autonomy
The right to be left alone
The right to a private life
The right to control information about oneself
The right to limit accessibility
The right of exclusive control of access to private realms
The right to minimize intrusiveness
The right to expect confidentiality
The right to enjoy solitude
The right to enjoy intimacy
The right to enjoy anonymity
The right to enjoy reserve
The right to secrecy

Table 2.1: *Privacy Interests.* David Flaherty lists thirteen aspects that repeatedly appear in privacy literature when describing information privacy [119].

a state of protection without having to actively pursue it. All else being equal, users undoubtedly would prefer systems without passwords or similar access control mechanisms, as long as they would not suffer any disadvantages from this. Only if any of their files are maliciously deleted or illegally copied, users will regret not having any security precautions in place. So what would be the analogy to a "break-in" from a privacy point of view?

Gary T. Marx, professor emeritus for sociology at MIT, has done extensive research in the areas of privacy and surveillance, dentifying *personal border crossings* as a core concept: *"Central to our acceptance or sense of outrage with respect to surveillance . . . are the implications for crossing personal borders"* [233]. Marx differentiates between four such border crossings that are perceived as privacy violations:

- *Natural borders:* Physical limitations of observations, such as walls and doors, clothing, darkness, but also sealed letters, telephone calls. Even facial expressions can form a natural border against the true feelings of a person.

- *Social borders:* Expectations about confidentiality for members of certain social roles, such as family members, doctors, or lawyers. This also includes expectations that your colleagues will not read personal fax messages addressed to you, or material that you left lying around the photocopy machine.

- *Spatial or temporal borders:* The usual expectations of people that parts of their life, both in time and social space, can remain sepa-

rated from each other. This would include a wild adolescent time that should not interfere with today's life as a father of four, or different social groups, such as your work colleagues and friends in your favorite bar.

- *Borders due to ephemeral or transitory effects:* This describes what is best known as a "fleeting moment," an unreflected utterance or action that we hope gets forgotten soon, or old pictures and letters that we put out in our trash. Seeing audio or video recordings of such events later, or observing someone sifting through our trash, will violate our expectations of being able to have information simply pass away unnoticed or forgotten.

Whenever personal information crosses any of these borders without our knowledge, our potential for possible actions – our decisional privacy – gets affected. When someone at the office suddenly mentions family problems that I have at home, or if circumstances of our youth suddenly are being brought up again even though we assumed that they were long forgotten, we perceive a violation of our local, informational, or communication privacy. This violation is by no means an absolute measure, but instead depends greatly on the individual circumstances, such as the kind of information transgressed, or the specific situation under which the information is disclosed. The effects such border crossing have on our lives, as well as the chances that they actually happen, are therefore a highly individual assertion.

2.1.3 Motivating Privacy

It is far from clear whether and to what extend society should support the individual with respect to his or her local, informational, decisional and communication privacy. Statements by Scott McNealy, president and CEO of Sun Microsystems, pointing out that *"you have no privacy anyway, get over it"* [372], as well as Peter Cochrane's editorial in Sovereign Magazine (when he was head of BT Research) claiming that *"all this secrecy is making life harder, more expensive, dangerous and less serendipitous"* [67], are representative for a large part of the population that fails to see the point of such seemingly paranoid secrecy.

In his book "Code and other Laws of Cyberspace" [217], Harvard law professor Lawrence Lessig tries to discern possible motivations for

having privacy in today's laws and social norms. He lists four major driving factors for privacy:

- *Privacy as empowerment:* Seeing privacy mainly as informational privacy, its aim is to give people the power to control the dissemination and spread of information about themselves. A recent legal discussion surrounding this motivation revolves around the question whether personal information should be seen as a private property (which would entail the rights to sell all or parts of it as the owner sees fit) or as a "moral right" (which would entitle the owner to ascert a certain level of control over her data even after she sells it) [302].[6]

- *Privacy as utility:* From the data subject's point of view, privacy can be seen as a utility providing more or less effective protection from nuisances such as unsolicited calls or emails. This view probably best follows Brandeis' "The right to be let alone"-definition of privacy, where the focus is on reducing the amount of disturbance for the individual.

- *Privacy as dignity:* Dignity can be described as "the presence of poise and self-respect in one's deportment to a degree that inspires respect." [264] This not only entails being free from unsubstantiated suspicions (for example when being the target of a wire tap, where the intrusion is usually not directly perceived as a disturbance), but rather focuses on the *balance* in information available between two people: analogous to having a conversation with a fully dressed person while being naked oneself, any relationship where there is a considerable information imbalance will make it much more difficult for those with less information about the other to keep their poise.

- *Privacy as constraint of power:* Privacy laws and moral norms to that extend can also be seen as a tool for keeping checks and balances on a ruling elite's powers. By limiting information gathering of a certain type, crimes or moral norms pertaining to that type of information cannot be effectively enforced. As Stuntz puts it: "Just as a law banning the use of contraceptives would tend to encourage bedroom searches, so also would a ban on bedroom

[6] See also our discussion on privacy as property in section 3.2.1 on page 70 below.

> searches tend to discourage laws prohibiting contraceptives" (as cited in [217]).

Depending upon the individual driving factor, an individual might be more or less willing to give up part of his or her privacy in exchange for a more secure life, a better job, or a cheaper product. The ability of data protection laws and regulations to influence this interplay between government and citizen, between employer and employee, and between manufacturer or service provider and customer, creates a social tension that requires a careful analysis of the underlying motivations in order to balance the protection of the individual and the public good. An example of how a particular motivation can drive public policy is the latest anti-spam legislation that recently passed both in Europe and in the US, which provides privacy as an utility by restricting the unsolicited sending of e-mail [240, 374]. In a similar manner, in March 2004 the Bundesverfassungsgericht (the German Supreme Court) ruled that an 1998 amendment to German's basic law enlarging law enforcements access to wire-tapping ("Der Grosse Lauschangriff") was unconstitutional, since it violated human dignity [323].

A good example for this tension between the public good and the protection of the individual can be found in the concept of communitarianism. Communitarians like Amitai Etzioni, professor for sociology at the George Washington University in Washington, D.C., and founder of the Communitarian Network, constantly question the usefulness of restricting society's power over the individual through privacy laws, or more general, to

> articulate a middle way between the politics of radical individualism and excessive stateism [111].

In his 1999 work "The Limits of Privacy" [111], Etzioni gives the example of seven-year-old Megan Kanka, who in 1994 was raped and strangled by her neighbor Jesse Timmendequas. No one in the neighborhood knew at that time that Timmendequas had been tried and convicted of two prior sex offenses before, and had served six years in prison for this just prior to moving in next to the Kankas. Megan Kanka's case triggered a wave of protests in many US American states, leading to virtually all states implementing some sort of registration law for convicted sex offenders, collectively known as "Megan's Law." Depending on the individual states, such registration procedures range

from registering with the local police station upon moving to a new place, to leaving blood and saliva samples or even having to post signs in one's front yard reading "Here lives a convicted sex offender"[7] [322]. While many criticize Megan's Law for punishing a person twice for the same crime (after all, the prison sentence has been served by then – the perpetual registration requirement equals a lifelong sentence and thus contradicts the aim of resocialization), others would like even more rigorous surveillance (e.g., with the help of tracking foot cuffs) or even a lifelong imprisonment in order to prevent any repeated offenses.[8] A similar lifelong-custody mechanism passed in 2004 a public referendum in Switzerland: Before being released from their prison sentence, psychologists will have to assess a sex offender's likelihood for relapse. Those with a negative outlook will then be taken directly into lifelong custody. Given the recent terrorist activity in many western democracies, many citizens might think the price of individual freedom that is made possible through rigorous privacy laws is possibly a price too high.

But it is not only violent crimes and homeland security that makes people wonder whether the effort spent on protecting personal privacy is worth it. Especially mundane everyday data, such as shopping lists or my current location – things that are very much publicly accessible, in contrast to, say, my diary, or my bank account balance and transactions – seem to have no reason for protection whatsoever. In many cases, collecting such data means added convenience, increased savings, or better service for the individual: using detailed consumer shopping profiles, stores will be able to offer special discounts, send only advertisements for items that really interest a particular customer, and provide additional information that is actually relevant to an individual. And, as Lessig remarks, any such data collection is not really about any individual at all:

> [N]o one spends money collecting these data to actually learn anything about you. They want to learn about people *like* you [217].

What could be some of the often cited dangers of a transparent society

[7] In May 2001, a judge in Texas ordered 21 convicted sex offenders not only to post signs in their front yards, but also place bumper stickers on their cars stating: "Danger! Registered Sex Offender in Vehicle" [322].

[8] Another problem with this approach is its broad application towards any "sex-offenses:" In some states, this also puts adult homosexuals or underage heterosexual teenagers having consentual sex on such lists.

then? What would be the harm in stores having comprehensive profiles on each of their customers in order to provide them with better services?

One potential drawback of more effective advertisement is probably the potential for manipulation: if, for example, I am identified as a mother of teenagers who regularly buys a certain breakfast-cereal, a targeted advertisement to buy a competitor's brand at half the price (or with twice as many loyalty points) might win the kid's favor, thus prompting me to switch to the potentially more expensive product (with a higher profit margin). Profiles also allow a process that sociologist David Lyon calls *social sorting* [228]:

> The increasingly automated discriminatory mechanisms for risk profiling and social categorizing represent a key means of reproducing and reinforcing social, economic, and cultural divisions in informational societies [227].

Since a small percentage of customers (whether it be in supermarkets or when selling airline tickets) typically makes a large percentage of profits,[9] using consumer loyalty cards or frequent flyer miles allows vendors to more accurately determine whether a certain customer is worth fighting for, e.g., when having to decide if a consumer complaint should receive fair treatment.

This might not only lead to withholding information from customers based on their profiles, but also to holding this information *against* them: When 59-year old Ron Rivera slipped on spilled yogurt in a Vons Supermarket in 1998 and subsequently sued for damages, the supermarket's management allegedly threatened to disclose Rivera's shopping profile, indicating that he was regularly buying more than average quantities of hard liquor and thus probably slipped because he was an alcoholic, rather than due to the yogurt [339]. In a similar incident, a husband's preference for expensive wine that was well documented in his supermarket profile, allowed his wife to claim a higher alimony after having subpoenaed the profile in court. Even if such examples pale in comparison to the huge number of transactions recorded everyday worldwide, they nevertheless indicate that this massive collection of mundane everyday facts will further increase through the use of ubiquitous computing, ultimately adding a significant burden to our lives, as Lessig explains:

[9] [148] cites IBM-analyst Merlin Stone with saying "In every sector, the top 20% of customers give 80% of the profit."

> The burden is on you, the monitored, first to establish your innocence, and second, to assure all who might see these ambiguous facts, that you are innocent [217].

This silent reversal of the classical presumption of innocence can lead to significant disadvantages for the data subject. An example for the sudden significance of these profiles is the fact that shortly after the September 11 attacks, FBI agents began collecting the shopping profiles and credit card records of each of the suspected terrorists in order to assemble a terrorist profile [21].[10] First reports of citizens who were falsely accused, e.g., because they shared a common name with a known terrorist [375] or had a similar fingerprint [222], give an example of how difficult it will be for an individual to contest findings from computerized investigative tools. Next generation profiling tools such as the airport security system CAPPS II[11] would be able to integrate such profiles in real-time, thus exacerbating this problem even further,[12] according to David Sobel, legal counsel at the Electronic Privacy Information Center (EPIC) and an expert for cryptography and privacy:

> Looking ahead to the CAPPS II system, that system will likely have access to a broad pool of information that is unlikely to be completely accurate. We will see an exponential increase in the number of people who will encounter these problems [375].

Complete transparency, however, can also help curb governmental power substantially, according to David Brin, author of the book "The Transparent Society" [48]. In his book, Brin argues that loosing our privacy can ultimately also have advantages: While up to now, only the rich and powerful had been able to spy on common citizens at

[10] Interestingly enough, the main shopping characteristic for all of the suspected terrorists wasn't a preference for hummus, but rather a tendency to order home-delivery pizza and paying for it by credit card.

[11] CAPPS stands for *Computer Assisted Passenger Pre-Screening System*. According to plans of the US American Transportation Security Administration (TSA), CAPPS II would assign a color-code to each flight passenger upon check-in, classifying the customer according to his or her security risk: Green for no danger; Yellow for potential danger that requires additional security checks; and Red for immediate danger that prompts alerting security personnel and denying boarding [101]. According to numbers from the TSA, up to 3-4 percent of all passenger would fall into the yellow category, and up to 500 passenger per year would be flagged as code red [168]. The categorization would take into account all public and commercial databases available.

[12] The Department of Homeland Security officially discontinued the CAPPS-II program after continued criticism in July 2004 [152].

will, the next technology would enable even ordinary individuals to "spy back," to "watch the watchers" in a society without secrets, where everybody's actions could be inspected by anybody else and thus could be held accountable, where the "surveillance" from above could now be counteracted by "sousveillance" from below [229].

Critics of Brin point out that "accountability" is a construct defined by public norms and thus will ultimately lead to a homogenization of society, where the moral values of the majority will threaten the plurality of values that forms an integral part of any democracy, simply by holding anybody outside of the norm "accountable" [217].

Summarizing, we can see that the ideal level of privacy can have very different realities, depending on the technically feasible and the socially desirable. The issues raised by the authors above and their colleagues are as follows:

- *Feasibility:* What can technology achieve (or better: prevent)? All laws and legislation require enforceability. If privacy violations are not traceable, the much stressed point of accountability (as developed in the fair information practices) becomes moot.

- *Convenience:* The advantages of free flow of information outweighs the personal risks in most cases. Only highly sensitive information, like sexual orientation, religion, etc might be worth protecting. Semi-public information like shopping habits, preferences, contact information, even health information, might better be publicly known so that I can enjoy the best service and protection possible.

- *Communitarian:* Personal privacy needs to be curbed for the greater good of society (trusting the government). Democratic societies may choose to appoint trusted entities to oversee certain private matters in order to improve life for the majority.

- *Egalitarian:* If everybody has access to the same information, it ceases to be a weapon in the hands of a few well-informed. Only when the watchers are being watched, all information they hold about me is equally worth the information I hold about them. Eventually, new forms of social interaction will evolve that are built upon these symmetrical information assets.

2.2 Privacy and Ubiquitous Computing

Privacy and data protection was always closely related to the technically feasible. At the end of the 19th century, it was the invention of modern photography that prompted Warren and Brandeis to rethink the concept of legal privacy protection. At the beginning of the 20th century, laws had to be reinterpreted again to take into account the possibilities of modern telecommunication (again, then supreme court judge Brandeis played a large part in that). And in the 1960s and 1970s, it was the implementation of efficient government through the use of modern databases that required yet another update of privacy laws, resulting in the first of today's modern data protection laws with their focus on data self-determination. In each instance, technology changed what was possible in the everyday and thus prompted – if sometimes with a considerable delay – a realignment of our notion of privacy.

After the commercialization of the Internet in the 1990s had initiated the last round of updates,[13] the dawn of ubiquitous computing promises the next revolution of "smart things." Even though many ubiquitous computing visions sound like AI-revisited, applications like the "intelligent car," or the "smart home" might not face the same fate as the dreams of intelligent machines that some 20 years ago researchers thought of being just around the corner. Ubiquitous computing often solves a much more mundane yet important problem, namely crossing media boundaries [120].

Using miniature sensors, cheap microchips, and wireless communication, computer technology can penetrate our everyday lives in a completely unobtrusive manner. Similarly, real world facts and phenomenons can be mapped on a computer with an unprecedented reliability and efficiency. The boundary between the real and virtual world seems to disappear – just like in a huge simulation it will be possible to model real world facts in real-time on a computer system.

Data protection and privacy is all about these mappings: translating facts of the real world into bits of information that can be stored for later retrieval. Ubiquitous computing is about the digitalization of our lives in order to allow computer systems to automatically process them. It comes as no surprise that ubiquitous computing has the potential to

[13] See for example the Children On-line Privacy Protection Act (COPPA) of 1998 [114] or the Electronic Communications Privacy Act (ECPA) of 2000 [59].

yet again change our perception of privacy in a significant manner. This qualitative quantum leap can be traced along five aspects of ubiquitous computing systems: the collection scale, manner, and motivation, as well as the data types and the data accessibility.

2.2.1 Collection Scale

The conscious surveillance of the actions and habits of our fellow men is probably as old as mankind. In the "good old times," this kind of observation was typically done by our closest neighbors, which in turn often drove "non-compatible" people out into the large cities, in which the large number of citizens and their high fluctuation would render this classical method of direct social monitoring impractical.

With the rise of automated data processing, machines began to take over the role of the curious neighbor. At first only available to governments, automated data processing soon found its way into commerce, both times facilitating a much more efficient management by providing detailed population or inventory information. However, while our neighbors would quickly note anything *out* of the ordinary, machines were now employed to actually determine what *was* ordinary: Not the deviations of the norm were noticed and tracked, but the average citizen and his or her ordinary everyday.

With ubiquitous computing, real life monitoring – the surveillance of the ordinary – will extend beyond today's credit card transaction, telephone connection records, and Web server logs. Even without assuming a single homogeneous surveillance network like Orwell's Big Brother, the sheer applicability of ubiquitous computing technology in diverse areas such as hospitals and nurseries, kindergartens, schools, universities, offices, restaurants, public places, homes, cars, shopping malls, and elderly care facilities, would create a comprehensive set of data trails that could cover us anywhere we would go.

Especially the "always on" vision of ubiquitous computing– alleviating us from laboriously switching various devices on and off as everything "stands ready" to our attention, right when we need it – would drastically extend this coverage over time. Instead of the spotty trails that can be obtained through our Internet logs when we are on-line, say, after work for an hour or two, smart homes and intelligent environments will not be switched off at night or while we are gone for lunch. In fact, it might not be even possible to turn such devices off, as they would not

feature a corresponding on-off-switch, but would sleep most of the time to preserve energy and wake up on their own whenever something of interest to them would happen. Anywhere and anytime, from sunrise to sunset, from cradle to grave, 24 hours a day, seven days a week. As Grudin [144] points out, the actual *selection* of data that is captured and stored will at the same time significantly alter the *value* of that information: *"Anything that is recorded instantly achieves a potential pervasiveness and immortality that it did not have before... Anything that does not 'make the cut' ... is invisible to someone inspecting the digital record at a different location or time."*

2.2.2 Collection Manner

When little children play Hide and Seek, they often cover their eyes with their hands in the believe that if they cannot see, others will not see them in turn. While they will learn eventually that the principle of reciprocity does not hold in this case, this apparent childish belief is much more difficult to unlearn than we might want to believe. Even years after playing their last game of Hide and Seek, many will assume that if they cannot see anybody else around, their actions will go unnoticed.

In the old days, this principle of reciprocity was actually a reasonable approximation of the collection manner in which people's action were observed. Only when one was out in public, others were able to see and draw their inferences. Once we entered the sanctuary of our own homes or those of others, we were shielded from the prying eyes of the public. This dichotomy of public and private was closely associated with the realities of space – the architecture of walls, windows, and doors, or the natural environment of woods and dense thickets: The presence and quality of a physical boundary provided an immediate indicator of the (potential) quality of privacy.

With the rise of electronic transactions, day-to-day actions like talking to a friend (over the phone) or buying groceries (using a credit-card) became noticeable beyond such physical boundaries. The presence or absence of others was not a good approximation of privacy anymore, as the digital trace of a transaction could be observed, stored, and retrieved from potentially anywhere in the world.

The deployment of ubiquitous computing technology will make it even more difficult to differentiate between public and private actions:

As ubiquitous computing tries to hide the use of technology, to make computers practically invisible, the level of awareness for such electronic transactions will drop drastically from today's implicit awareness through the use of physical tokens such as credit-cards or mobile phones. In a fully computerized environment, potentially any item could take fingerprints and wire them halfway around the world, take pictures, measure body temperature, or observe one's gait in order to draw far-reaching conclusions about a persons physical and mental conditions. Neither data collection nor continuous surveillance activities will have recognizable markers that would indicate the publicity of actions – ultimately requiring us to assume that at any point in time, in any location, any of our actions could potentially be recorded electronically and thus made public.

2.2.3 Data Types

With ubiquitous computing, also the type of information that is collected will change. The village gossip was based on the observation of neighbors and fellow citizens and on a person's discussions with others. This information was by definition "soft" information, that is, it was based on an individuals personal reception and more often that not, two different people observing the same fact would retell widely different accounts of it. While this would often result in rather exaggerated claims, it nevertheless retained some level of deniability.

Modern data processing seems far away from the village gossip of old. It concerns itself with "hard" information – with facts, rather than hearsay. Instead of capturing the individual (and error-prone) human perception, it collects "true" information such as names, birth dates, addresses, income levels, or lists of purchases. Using statistical models, this information can subsequently be used to draw inferences on a person's life based on his or her residence and shopping preferences.

Ubiquitous computing will extend this selection of hard facts beyond traditional information types: smart shirts and underwear will be able to record health data such as blood pressure, heart rate, perspiration, or glucose levels in real time; smart supermarket shelves will not only know what items a person bought, but also in what order and how long he or she hesitated before reaching out; mobile phones with GPS-locator allow friends and family to know one's whereabouts at anytime, unless one decides to turn the service off and find a good excuse for

doing so.

Data mining technology will allow researchers, politicians, and marketers to make sense of this ever increasing stream of minute details, by correlating widely disparate information such as chocolate consummation and shower habits (for example to infer the beginning of a new relationship), and through comparing information from hundreds of similar people in order to discern population patterns. This has also significant implications for the anonymization of such data, as perceived information such as one's location over the course of a day, or the particular way of walking as registered by floor pressure sensors, or one's individual breathing pattern, might turn out to be easily identifiable even if collected completely anonymous.[14]

With a wide array of new sensors and collection mechanisms, ubiquitous computing technology could potentially allow inferring the "soft" gossip of old based on the "hard" facts of today, thus not only giving it new credibility (by being based on facts, not hearsay) but also eventually incapacitating our own judgments about personal beliefs and feelings based on computerized self-assessments, e.g., inferring our emotional attachment to our partner based on our heart-rate and eye blinking rate.

2.2.4 Collection Motivations

As we have seen in the previous chapter on privacy and its motivations, incentive (i.e., the "Why?") plays an important role when it comes to facilitating or preventing data collection. And just as the reasons for wanting privacy have changed over the years, so have the motivations for collecting this data.

Our neighbor's eyes and ears looked for the unusual, the out-of-ordinary events that would make for attractive gossip. Consequently, people who were adept at "blending in", those who hardly attracted attention due to their ordinary lives and average physical features, would get the least scrutiny.

With automated data processing, attention shifted from the unusual to the ordinary: Governments tried to make better policies by having better data on whom they governed, and that meant finding out what the average citizen did, liked, or feared. Companies tried to find

[14] See the work of Sweeney [331] and Beresford and Stajano [35], which we also discuss in sections 3.4.3 and 5.2.3 below.

out what goods consumers wanted (or did not yet know they wanted). Questionnaires were used to (and still are) solicit the preferences of the masses, in order to better understand what products would work and which would not. With modern data analysis methods, large amounts of statistical information, such as family income, street address, or political preferences, can be statistically correlated in order to segment population groups and predict human behavior (e.g., a family moving into the suburbs might soon decide to buy a lawn mower, as most families there own one).

Providing better services and/or better products will still be at the heart of many future ubiquitous computing systems, yet what data is necessary to predict this becomes less and less clear, as more different types of information can be collected (see section 2.2.3 above). With better data mining capabilities than ever before, virtually anything can be of importance, if only enough statistical data on it can be collected. *Context-Awareness* is one of the main paradigms in ubiquitous computing, as it is thought to enable otherwise "dumb" systems to predict the user's needs and intents without involving any "real" intelligence. Not surprisingly, the more such context information is available, the better these systems are expected to perform. Instead of targeted data collections of specific information for a certain purpose, future ubiquitous computing systems could easily attempt to collect *any* and *all* information possibly available, thus maximizing their chances for correctly determining the user's context from it.

2.2.5 Data Accessibility

Information is only of worth if one can find it: collecting large amount of data without having efficient retrieval mechanisms in place suggests not collecting it in the first place. In the old days, retrieving gossip was typically limited to a particular village or neighborhood. By moving into a different town or even into a larger, anonymous city, the previously assembled body of "knowledge" would typically be rendered inaccessible for the newly acquired neighbors, requiring them to start out anew.

With modern information networks, information can travel at nearly light speed around the globe, and modern database management systems allow for the efficient retrieval of minute details out of huge, federated databases from a wide variety of sources. However, even though

standardized interface definitions exist, integrating these sources is far from a trivial problem, as the large number of failed data-integration projects in both government and industry have shown.[15]

In the vision of ubiquitous computing, such kind of information systems would not be primarily designed with humans in mind (and thus lead to often non-interoperable system), but directly target machine-to-machine interactions: Smart things would "talk" to other smart things in order to collaboratively determine the current context, and large networks of autonomous sensor nodes would send sensor readings back and forth in order to arrive at a global state based on hundreds of individual sensor readings. Similarly, improved human-computer interfaces would allow easy access to non-traditional data formats such as video- and audio-streams, e.g., for automated diary applications that would document one's everyday in a continuous multimedia format. Living in a world of smart cooperating objects, the "freedom of movement" for personal information would be greatly increased, both between humans and computers (How well can I search your memory?) and between cooperating artifacts (What is my artifact telling yours?).

2.3 Summary

This chapter has tried to frame the problem of personal privacy in ubiquitous computing from two different point of views: *What* is it that we mean by privacy? And *why* does ubiquitous computing affect privacy substantially?

We have seen that the concept of privacy can be approached from a number of angles, and that each definition typically focuses on a particular area – or facet – of privacy (section 2.1.1). Looking at its procedural facets, one can distinguishing between bodily, territorial, informational, and communication privacy. Its functional facets divide privacy into zonal, relational, and decisional privacy. And its constitutional facets are comprised of solitude, anonymity, and control.

In addition to trying to understand the (abstract) *concept* of privacy, we have also explored the *situations* in which one might feel that his or her privacy has been violated (section 2.1.2). We now know that

[15] A 1996 project of FleetBoston Financial Corp. tried to pull together customer information from over 60 source systems, but was practically terminated after three years due to the underestimated complexity of reconciling and integrating the data sources [97]. According to the Standish Group, 88 percent of data integration projects fail or overrun their target budgets by an average of 66 percent [282].

unanticipated data flows across personal *borders* – such as physical barriers (e.g., doors, letters), social barriers (i.e., different peer groups), distances over time and space, and fleeting moments – will typically be perceived as privacy invasive. Building and deploying ubiquitous computing systems will often facilitate crossing such borders.

Yet we have also learned that whether or not to prevent such borders is a highly disputed topic. Section 2.1.3 listed both public safety and personal security reasons as important driving factors for *less* privacy, rather than more. In order to decide what kind of privacy we want to expect from future ubiquitous computing environments, these values will ultimately have to be agreed upon by society.

The second part of this chapter focused on the qualitative differences a world full of smart things would have in terms of privacy (section 2.2). It identified five such qualitative differences: an increased collection scale, a more subtle collection manner, new types of data, a higher collection motivation, and improved data accessibility and exchange.

The following chapter will provide us with the knowledge about the mechanisms at our disposal to react to this qualitative difference. It will describe social, legal, and technical mechanisms readily available to us that we will be able to rely upon in chapter 4 later, in order to ground our technical privacy-awareness infrastructure in moral norms, legal requirements, and technical possibilities.

3 Privacy Mechanisms and Principles

I never thrust my nose into other men's porridge.
Don Quixote[1]

The last chapter focused on the *What* of privacy. This one will focus on the *How*. We will visit three different areas from where we will draw support for our envisioned privacy-awareness system for ubiquitous computing: social, legal, and technical mechanisms, and use them to develop a set of guiding principles along which we will develop our technical infrastructure.

Social mechanisms seem to be the least relevant to our task, as they are often elusive and hard to describe (as we have seen with the concept of privacy in the previous chapter), and thus seem to be rather difficult to employ directly. However, by realizing which social factors govern our interactions with our neighbors and fellow citizens, we can focus our software and hardware development efforts on the technically feasible, and use existing social mechanisms to support our system in areas where technology alone will be inadequate.

Legal mechanisms are a codification of social norms, and can thus be much better practically applied. However, as Lawrence Lessig pointed out in his book *Code is Law* [217], their effectiveness depends to a large extend on proper enforcement, which in a ubiquitous computing future would certainly be contingent on the technical implementation of systems. We will be looking at different legal frameworks for privacy protection around the world in order to assess not only what kind of privacy protection these laws encode, but also how we can build our system to support such laws.

Technical mechanisms will form our building blocks for our system, and this chapter briefly introduces them so we can readily employ them in our design in chapter 4: encryption and authentication mechanisms

[1] In Miguel de Cervante's "Don Quixote."

for secure communication; transparency and trust mechanisms to communicate privacy information to the user; and anonymity and pseudonymity mechanisms to minimize data collection whenever possible. Much of this technology has been developed for Web privacy, so we will take various parts of these tools and reassemble them in our ubiquitous computing privacy infrastructure to suite the changing requirements of an "Internet of Things."

3.1 Social Mechanisms

Since the term *social tools* might be misleading, as it has been increasingly used to describe software that facilitates social interactions, e.g., instant messaging or blogging [45], we use the term *social mechanisms* in order to describe tools, methods, and procedures that exist beyond the codification of laws and the implementation of technical infrastructures.

In particular, we want to briefly look into ethics and trust issues. Ethics is relevant for our privacy discussion because it teaches right from wrong, good from bad, and thus has a direct influence on how we judge privacy violations, or value our privacy and the privacy of others. Trust, then, is the next step after having made up our minds on what to do, as we will need to make assumptions about the actions (or inactions) of others in order to justify our own actions (or inactions).

Both ethics and trust might not be directly usable when trying to build privacy-aware ubiquitous computing infrastructures. However, by looking at social science research surrounding these two concepts, we hope to learn two things: What influences human behavior when it comes to privacy issues? And how do these issues limit or facilitate what we can achieve with laws or technologies?

3.1.1 Ethics

The field of ethics, also called *moral philosophy*, has its roots in the classical work of the Greek philosophers, such as Socrates, Plato, and Aristotle, who were contemplating the proper ways to lead a "good life" in pursuit of "true happiness" [280]. Modern western philosophers (where modern designates anything following the 16th century) such as Thomas Hobbes, David Hume, and Immanuel Kant, rediscovered these questions in the Renaissance – a time when the creation of modern

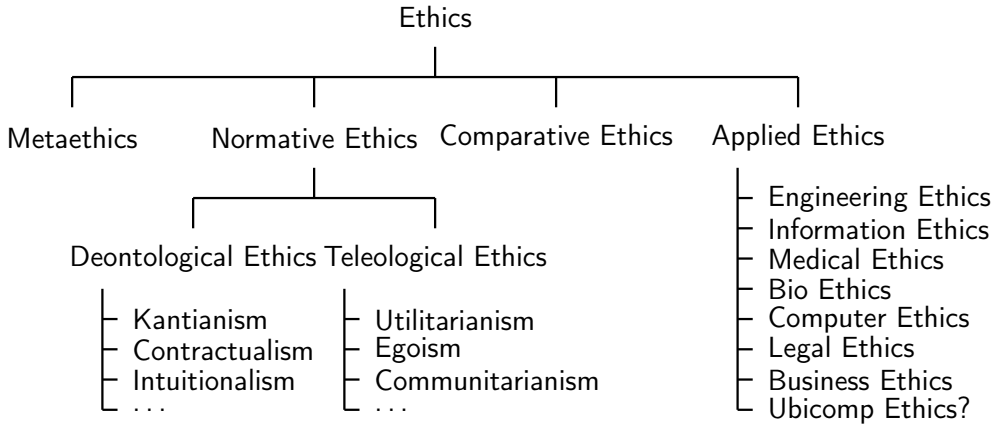


Figure 3.1: *Ethics Overview.* The field of ethics, or moral philosophy, can roughly be divided into four subfields: metaethics, normative ethics, comparative ethics, and applied ethics.

nation states, the Reformation of Luther and Calvin, and the scientific discoveries of Kepler and Newton designated the end of the dark Middle Ages [312].

The word "ethics" comes from the Greek *ethos*, which means "disposition" or "character." The word "morality", in turn, comes from the Greek *mores*, which means "social rules" or "customs". Today, these meanings are often reversed, with morality reflecting one's personal beliefs (which are really governing the behavior of individuals), and ethics referring to the external "science" of moral values (i.e., the theories) [136].

Ethics Theories

Figure 3.1 gives a (very much) simplified overview of the large and complex field of ethics. Research in ethics can roughly be broken into four different subareas: *analytical* ethics (often called "metaethics"), *normative* ethics, *descriptive* ethics (also called "comparative ethics"), and *applied* ethics.

Metaethics concerns itself with the nature of ethical statements, such as whether moral values are eternal truths or simply human conventions,[2] or if and why humans need moral values. While seemingly of less practical value, metaethical theories can play a vital role in con-

[2]Plato compared moral values to mathematics: just as $1+1=2$ is a universal truth that cannot be altered by humans, Plato saw moral principles such as "murder is wrong" as absolute and eternal.

temporary ethical problems, for example regarding the right to privacy for sex offenders, as discussed in section 2.1.3, where questions such as "where do rights come from?" and "what kind of people have rights?" are of high relevance.

Normative ethics tries to work out what these moral values should be, or how a certain moral standard is to be evaluated. This is probably the most "popular" area of ethics, and a wealth of different theories exist on what is (or ought to be) morally good and bad. These theories are typically either called *deontological* or *teleological*.

Deontological ethics (*deon* is Greek for "duty", *logos* means "science") infers moral obligations from the characteristics of a certain action, without regard for its consequences. Thus, an action that is morally good might still have serious negative consequences. One of the most prominent proponents of deontological ethics was Immanuel Kant, who formulated a "Golden Rule," his *categorical imperative*, for determining the morality of an action:

> Act only on that maxim whereby thou canst at the same time will that it should become a universal law [189].

Note that Kant's Golden Rule is not just a reformulation of the Biblical Golden Rule "All things whatsoever you would have men do unto you, do you ever so to them,"[3] as it explicitly requires moral principles to be universally applicable, to become a universal law of nature. It is thus a *categorical* imperative, not just a *hypothetical* imperative, which only applies conditionally (e.g., only if you want people to do A to you, do A to them).

Other important deontological theories are *contractualism* and *intuitionalism*. Intuitionalists such as William Ross tried to counter the critique of Kantian ethics being vacuous (as the categorical imperative never explicitly says what should be done), and specifically listed six duties that should be given independent weight: fidelity, reparation, gratitude, beneficence, non-maleficence, and self-improvement [320]. If an act falls under one of these obligations, it ought to be carried out. If two or more competing duties exist, intuition would need to tell us which obligation would override the other.

[3] In (Matthew 7:12). Similar rules can also be found in Jewish philosophy (e.g., Hillel, 1st century BC) or from Eastern philosophers such as Confucius.

Contractualists, on the other hand, follow the tradition of Thomas Hobbes and assume a *social contract* as the basis for any moral decision. According to Hobbes, all of man's voluntary acts are aimed at self-pleasure or self-preservation [320]. This leads to a selfish "war of all against all," which can not be solved by appealing to any morality, but only through reason: by entering a social contract, we would give up our rights to attack others in turn for their giving up their rights to attack us. This is a situation analogous to the famous Prisoner's Dilemma [87], a popular example in both social sciences and economics that illustrates the limits of pure rational choice. Only if all "players" cooperate (e.g., enter a social contract and give up some of their rights), an optimal "payoff" can be found, even though the dominant strategy[4] for each player would be to default on the others.

Whether it is Kantianism, Intuitionalism, or Contractualism – deontological theories stipulate that doing ones duty is morally right, and that duties can be reasoned out by deriving them from moral truths. Their focus is on intention, not outcome. In the area of privacy, this would amount to a view of privacy as a basic right that is non-negotiable. Coming back to the example of section 2.1.3, a deontological view might grant a released sex offender the right to remain anonymous, even if the possibility of a relapse would threaten children in their new neighborhood. Their "right to a second chance" after debts to society have been paid (i.e., their prison sentence had been served) would apply independently of their actual crime.

Teleological ethics on the other hand, derives morality not from the intentions, but from the *consequences* of actions, e.g., whether it leads to "desirable" effects (*telos* is Greek for "goal" or "end"). In the context of privacy, this would allow for example a supermarket to overlook personal shopping habits if it would provide consumers with shopping recommendations that could save them money (e.g., by pointing out sales). The exact nature of these effects, i.e., what exactly constitutes a desirable effect, is of course no less debated than the moral truths of the deontologists.

The most prominent teleological ethical theory is that of *utilitarianism*. Its main proponents were the late 18th- and 19th-century English philosophers Jeremy Bentham and John Stuart Mill. Its central insight

[4] A *dominant strategy* yields the highest payoff of all your available strategies for every choice the other player or players make [87].

is that one ought to promote happiness and prevent unhappiness whenever possible [353]. Bentham was an ardent promoter of legal and social reforms in his days, and devised utilitarianism as the moral principle on which to base such reforms. To Bentham, the greatest good was whatever policy would cause the greatest happiness for the greatest numbers [363].[5] He proposed a *Hedonic Calculus*,[6] which would allow anybody to actually calculate the amount of happiness any action might cause (and thus its degree of moral rightness). Using seven "vectors" of pleasures and pains (intensity, duration, certainty, propinquity,[7] fecundity,[8] impurity,[9] and extend), one would add up the pros and cons for each individual involved and weigh them against each other [357].

Bentham's probably most "famous" contribution to today's privacy discussion is the *panopticon* – a model prison run in a hitherto unprecedented economical fashion:

> [The architecture] incorporates a tower central to an annular building that is divided into cells, each cell extending the entire thickness of the building to allow inner and outer windows. The occupants of the cells ... are thus backlit, isolated from one another by walls, and subject to scrutiny both collectively and individually by an observer in the tower who remains unseen [28].

Bentham envisioned his concept to appeal not only to prisons, but to hospitals, schools, and factories as well. Not only would the central design keep staffing levels low, but since no one would be able to tell whether or not he or she was under watch, everybody would exercise self-discipline brought on by the uncertainty of being under surveillance [125].[10]

John Stuart Mill, son of Bentham's fellow utilitarian James Mill, actually first coined the term "utilitarianism" in his similarly named 1861 article [242]. Mill disagreed with Bentham over the ability to calculate

[5] This principle was originally proposed by Francis Hutcheson in his "Inquiry into the Original of our Ideas of Beauty and Virtue" (1725) where he says *"That action is best which procures the greatest happiness for the greatest numbers."* Bentham later dropped the second qualification and spoke only of "the greatest happiness principle" [363].

[6] Also often called *felicific calculus*.

[7] Proximity, nearness.

[8] Prolificacy, fertility, i.e., the probability it has of being followed by sensations of the same kind.

[9] In Benthams sense: the probability it has of being followed by sensations of the opposite kind.

[10] While these proposals were a positive contribution to a much needed prison reform in Bentham's time, today's privacy discussions usually cite them as an example of excessive and inhumane surveillance.

this utility, saying that happiness should not merely be assessed by quantity, but by quality as well.

J.S. Mill is often associated with the idea of decisional privacy, as he was an ardent proponent of the freedom of individuals from government interference. In his 1859 essay *On Liberty*, Mill proposed as the proper balance between individual liberty and governmental authority the "harm principle:"

> [T]he only purpose for which power can be rightfully exercised over any member of a civilized community, against his will, is to prevent harm to others [241].

Even if utilitarianism at first seems to stand in opposition to liberalism, as the greatest happiness of the greatest numbers could potentially be best achieved under a dictatorship,[11] both Bentham and Mill see the individual as the best judge for his or her own happiness, thus suggesting it is best to leave people free to make their own choices.

Ethical *egoism* takes liberalism one step further, as it views morality as primarily concerned with the well-being of the individual. It considers the morality of actions not universally (i.e., for all of mankind, or at least for a larger group of people), but only with respect to an individual's interests: *"everybody should be doing what is in her or his own interest"* [320]. Ultimately, this often turns into a form of indirect utilitarianism, stipulating that all will be better off if everybody just follows their own interests. This follows roughly along the lines of Warren and Brandeis' "right to be let alone" (see section 2.1.1 above): keeping to ourselves is seen as the best recipe for protecting the privacy of all.

While the need for personal autonomy and individualism might have been crucial for the development of stable western democracy in the 18th and 19th century, modern *communitarianists* such as Amitai Etzioni [112] (c.f. section 2.1.3) or John Rawls [279] feel that in today's society, the pendulum has often swung too far. Communitarianism tries to seek a more balanced approach between individual rights and social responsibilities [111], feeling that strong US governmental privacy laws do not serve the common good and instead advocate a more "European" approach, where privacy is more of a contingent right, derived from deontological values but limited where it hinders public good. This is

[11] Both Orwell's *1984* and Huxley's *Brave New World* have often been said to be parodies on utilitarian societies.

countered by modern liberalists such as Sir Isaiah Berlin, who, in the tradition of Bentham and Mills, view the freedom to make moral choices as the most important freedom, consequently arguing that government should allow individuals the freedom to pursue their "own ideas"[12] [37].

The differences between US and European privacy morals, as well as between the UK and Continental Europe, are subject to *comparative* or *descriptive* ethics. Instead of trying to find a recipe for morally good living, comparative ethics investigates differences in ethical beliefs and values, as explained by physical and economic conditions, opportunities for cross-cultural contact, and inherited traditions.

Applied Ethics

While moral philosophers have always concerned themselves with practical questions,[13] it was not until the mid-1960s and the growing US civil-rights movement that the field of practical, or *applied*, ethics was established.

Applied ethics tries to apply findings in metaethics and normative ethics to concrete examples, such as equality (gender, race), environmentalism, war and peace, abortion, or genetics. In many cases, this resulted in the establishment of independent research areas, e.g., bioethics [266], medical ethics [31], computer ethics [183], information ethics [200], legal ethics [378], or engineering ethics [182].

Professional associations, such as the National Society of Professional Engineers,[14] or the Association for Computing Machinery (ACM),[15] try to give practical guidance to their members through publishing a *Code of Ethics* for their respective fields (see table 3.1). They are typically a mixture of deontological ("be honest and trustworthy") and teleological ("contribute to society and human well-being") approaches.

Applied ethics are especially relevant in the area of new technological advancement, such as genetic engineering, nuclear energy, or computing technology, such as ubiquitous computing:

> New technologies seem to pose ethical issues when they create new possibilities for human action, both individual action

[12] Berlin calls this a *negative liberty*: a freedom from restrictions on the individual in the tradition of Hobbes and Locke. This contrasts *positive liberty*, i.e., the freedom to act to fulfill one's own potential [359].

[13] Utilitarians such as Bentham and Mills were ardent proponents of legal and social reform, evident, e.g., in their concern for the penitentiary system (Bentham) or womens rights (Mills).

[14] See `www.nspe.org`

[15] See `www.acm.org`

1. *Contribute to society and human well-being.* When designing or implementing systems, computing professionals must attempt to ensure that the products of their efforts will be used in socially responsible ways, will meet social needs, and will avoid harmful effects to health and welfare
2. *Avoid harm to others.* To minimize the possibility of indirectly harming others, computing professionals must minimize malfunctions by following generally accepted standards for system design and testing. Furthermore, it is often necessary to assess the social consequences of systems to project the likelihood of any serious harm to others.
3. *Be honest and trustworthy.* Honesty is an essential component of trust. Without trust an organization cannot function effectively. The honest computing professional will not make deliberately false or deceptive claims about a system or system design, but will instead provide full disclosure of all pertinent system limitations and problems.
4. *Be fair and take action not to discriminate.* Discrimination on the basis of race, sex, religion, age, disability, national origin, or other such factors is an explicit violation of ACM policy and will not be tolerated.
5. *Honor property rights including copyrights and patent.* Violation of copyrights, patents, trade secrets and the terms of license agreements is prohibited by law in most circumstances. Even when software is not so protected, such violations are contrary to professional behavior.
6. *Give proper credit for intellectual property.* Computing professionals are obligated to protect the integrity of intellectual property. Specifically, one must not take credit for other's ideas or work, even in cases where the work has not been explicitly protected by copyright, patent, etc.
7. *Respect the privacy of others.* This imperative implies that only the necessary amount of personal information be collected in a system, that retention and disposal periods for that information be clearly defined and enforced, and that personal information gathered for a specific purpose not be used for other purposes without consent of the individual(s).
8. *Honor confidentiality.* The principle of honesty extends to issues of confidentiality of information whenever one has made an explicit promise to honor confidentiality or, implicitly, when private information not directly related to the performance of one's duties becomes available.

Table 3.1: *ACM Code of Ethics – Moral Imperatives* (excerpt from `www.acm.org/constitution/code.html`). Professional associations such as the ACM use Codes of Ethics to provide practical ethical guidance to their members. They are typically a mixture of deontological and teleological ethical approaches.

and collective or institutional behavior [183].

Many countries and organizations pursue ethical questions of new technologies in special *technology assessment* projects, often institutionalized, such as the US Office of Technology Assessment (OTA),[16] the Office of Technology Assessment at the German Parliament (TAB),[17] or the Institute for Prospective Technological Studies of the European Commission (IPTS).[18] They are chartered to research the impact of new technology on different sectors of society, as well as evaluate policy-relevant options which involve technology. Moore [247] calls these issues surrounding new technology, and in particular computer technology, "policy vacuums," a confrontation with choices about whether and how to pursue the opportunities new technology offers, without having an established set of policies on how to make these choices.

A straightforward approach to resolve and fill such policy vacuums would be to take the ethical principles and normative values found in the previous section, and apply them to the new situations created by new technology. However, as Moore points out:

> If we do not know what we are dealing with, we do not know which rules or principles should be applied [247].

Johnson [183] gives sending an e-mail message as an example, asking whether it is more akin to sending a postcard, having a phone conversation, or sending a letter? A similar example would be the recently introduced e-mail service from Google, Gmail,[19] which inserts ads into incoming e-mail messages, based on the actual message content [30]. While privacy advocates have asked Google to suspend the service [271], noting that the scanning of confidential email for inserting third party ad content would violate the implicit trust of an email service provider, others have compared inserting Google's ad-service with automated spam-blockers and virus-scanners – a computerized process matching certain byte-sequences, without any human intervention – and thus see Gmail no more privacy invasive than any other on-line e-mail service [259].

Lessig [217] gives a similar example in the area of ubiquitous computing and law enforcement: What if future smart home appliances

[16] In 1995, US Congress closed down the OTA for fiscal reasons [66]. The US government has been without independent technology assessment since.

[17] See `www.tab.fzk.de`

[18] See `www.jrc.es`

[19] See `gmail.google.com`

would by law be required to report any unlawful behavior, e.g., storing explosives in the fridge, watching Nazi propaganda videos,[20] or knife-stabbing humans?[21] Such a "spy in the kitchen" [172] would not actually leak any lawful information or action, only major crimes such as murder or terrorist activity. Just as Lessig asks which of the different possible motivations for privacy (empowerment, utility, dignity, or constraint of power)[22] would apply in this case,[23] we have to ask which conception of established moral values should apply in such a smart environment: A communitarian like Etzioni might judge the benefits of catching criminals worth the (for law-abiding citizen not even applicable) loss of privacy, while a libertarian such as Mill might see the privacy of the home essential to any development of decisional privacy, and thus democracy.

Figure 3.2 summarizes the above thought experiment, tracing the influence of our moral values and ethical principles down to the laws and regulations resulting from them, and ultimately to the technology created within the parameters set forth by a legislative body informed by independent technology assessment. We will revisit this interplay between ethics, laws, and technology in our discussion (section 3.2.4) below. But first, we want to briefly look at the social mechanisms at play when it comes to trust – both between people, and between people and machines. This will be important as we will need to rely on trust, whatever privacy architecture we are proposing. Knowing if and when such trust will evolve will allow us to put our technology on much firmer ground.

3.1.2 Trust

Just as privacy, the concept of trust has received considerable attention over the last years, mostly in social-science literature, such as psychology, sociology, or political science, but also in fields such as economy or sociobiology [221].

Contractarians like Hobbes or Locke, who assumed a social contract in order to impose moral behavior in people, required trust into a

[20] Which is a crime in Germany.

[21] Lessig's actual example uses a computer worm that scans the home computer and detects illegal software copies or classified documents [217].

[22] See Lessig's "driving factors for privacy" on page 30.

[23] A case in which the constitutional question of a "unreasonable search and seizure," as protected by the Fourth Amendment, would be challenged (as such "smart" search would not entail any of the burdens usually associated with a physical house search).

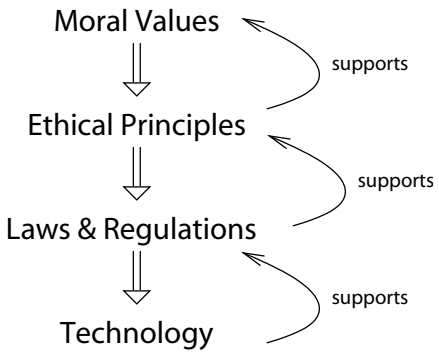


Figure 3.2: *Ethical Influence.* In an ideal world, our moral beliefs would be the basis for the ethical principles our community creates. These would be directly reflected in our laws and regulations, which in turn would govern the technology we create. In reality, all of these factors influence each other (see figure 3.3 on page 83).

sovereign authority and thus laid the groundwork for moral philosophers and political scientists to define trust primarily as a contractual element [158]. However, modern research began questioning this limitation, and extended the concept of trust beyond symmetric relationships and goods exchange. One of the earliest such definitions comes from Niklas Luhman, who already in 1968 defined trust as *"a mechanism to reduce social complexity"* [225]. Citing Worchel [379], Lewicki and Bunker group modern trust definitions and their corresponding research intro three areas [221]:

1. *Psychology:* Focusing on individual personality differences in the readiness to trust, trust is conceptualized as a belief, expectancy, or feeling that has its origins in the individual's early psychosocial development.

2. *Social and Political Scientists:* Focusing on trust as an institutional phenomenon, it is conceptualized as appearing both within and between institutions and organizations, and as trust that individuals put in those institutions.

3. *Economists:* Focusing on the interpersonal transactions between individuals, trust is conceptualized as an expectation in outcomes and as a risk-management when acting on such expectations.

For our purpose, all three areas are of importance, as each plays an important part in today's "information economy" [338], where we trade our personal information for services or other tangible (or intangible)

benefits.[24] Depending on each individual's disposition, one might feel more or less comfortable giving out personal information in a ubiquitous computing environment, and individual risk management strategies might be employed in order to decide under what conditions such information should be disclosed, given the amount of trust one places in both the data collector and the existing enforcement mechanisms (i.e., legal, social, and market forces). Trust in any of these areas thus forms an integral part of any technical privacy solution, for the following two reasons:

1. *Contractual Nature:* Releasing personal information to a third party requires trust, a minimum degree of faith that the receiving party will handle this information in the agreed manner. In his *Leviathan*, Hobbes calls this first stage of a contract, apart from the actual performance, the *covenant*, an exchange of promises which we will need to trust in [130].

2. *Institutional Trust:* As our personal data is typically collected by institutions, not individuals that we encounter face-to-face, a disclosure of private information requires trust in abstract entities such as corporations or the government, something that seems to have been in decline for years, as Robert D. Putnam succinctly described in his influential article *Bowling Alone* [274].

The following attempts to describe the concept of trust from each of the three areas given above – psychology, social sciences, and economics – and examines how our understanding of these aspects will affect our understanding of, and our trust or distrust in, any technical or non-technical privacy solution.

Psychology of Trust

Trust is a very personal issue, just as personal privacy is. And just as studies have repeatedly shown that different people have very different attitudes toward their privacy [84, 333], people typically handle their trust related decisions also quite differently, based on factors such as the current situation, the actors involved, their prior personal experience, and individual disposition [237].

Another, more subtle connection between privacy and trust can be seen at a perceptional level: Similar to privacy, trust corresponds to *"a*

[24] See also our discussion of privacy as personal property in section 3.2.1 on page 66.

certain feeling that is best perceived once it is missing, e.g., when moving from a friendly and 'secure' neighborhood into a tense and insecure one" [22]. Trust is very often an unconscious act, something Lagerspetz calls *Ex Post-Trust* [202], a retrospective realization once it is betrayed or once we realized what could have happened. Luhmann consequently uses the term *trust* only in cases where one is actually aware of one's own trust – in all other cases, when one is not consciously evaluating alternatives, Luhmann uses the term *confidence* instead [225]. Lagerspetz notes that *"the less I am aware of my trust, the stronger it seems to be,"* and that *"talking about trust already means considering the possibility of betrayal"* [202].

This elusiveness of trust has prompted some to conclude that trust is not something that one *does*, neither being a mental activity (such as a feeling) nor a plan (such as taking a risk), but instead that it is something that "lies in the eye of the beholder," i.e., that it is something that can be attributed only to a third person, not to myself, similar to attributes such as generosity, spontaneity, or innocence [202]. Baier offers a more practical definition:

> By trusting others, one grants them the option of violating this trust, yet at the same time expects them not to make use of it [23].

However, just as Luhmann differentiates between confidence and trust, Baier notes a difference between *relying* on others not to do something, and *trusting* them not to do something: By trusting someone we actually come to rely on their *benevolence* toward us – reliance does not need goodwill, as we might count, e.g., on fear from prosecution [23]. Baier sees trusting someone as giving them *discretionary powers* over something that is dear to me [23]. In order for a trust relationship to be of a certain permanence, Baier points out that the ability to "negotiate" these discretionary powers is essential, i.e., the ability to forgive on the side of the trusting party, as well as the ability to accept this forgiveness on the side of the trusted party.

Concepts such as forgiveness or ex-post trust are indicative of the hard-to-grasp nature of trust that makes it difficult to employ trust in a technical manner. This will become more apparent when we look at technical trust mechanisms in section 5.1.4. With all its elusiveness, however, trust is nevertheless an important part of any society that

can hardly be replaced with technology, as research in social sciences has shown.

Sociology of Trust

Sociologists and political scientists view trust not just as a personal fancy, but as an essential ingredient to a stable and prosperous society: *"Where trust and social networks flourish, individuals, firms, neighborhoods and even nations prosper"* [275].

The connection between trust and government can be traced back to Hobbes' concept of a supreme *sovereign*, in which citizens need to put their trust in order to collectively submit under his rule [130].[25] While first liberalism and later the concept of liberal democracy stipulated that one should be *less* trusting when it comes to governmental institutions,[26] contemporary social and political science views trust as a prerequisite for effective democracy [175].

Following Robert Putnam seminal article *Bowling Alone* [274], trust has since become *"perhaps the most essential part of* social captial*, ...features of social organizations ...that facilitate coordination and cooperation for mutual benefits"* [336]. Inspired by Alexis de Tocqueville's *Democracy in America* [231],[27] Putnam revived the term *social captial* to define *"features of social organization such as networks, norms, and social trust that facilitate coordination and cooperation for mutual benefit"* [274] that were *"domains of neither the state nor the market"* [88].

In his 1995 article, and in much more detail in his follow-up book [275], Putnam noted that participation in society in form of civic associations (e.g., church related groups) and good neighborliness had fallen sharply in the US, eroding the all-important social capital needed by democracies to function smoothly [274]. A 1996 poll by the Washing-

[25] British philosopher Onora O'Neill mentions in her 2002 Reith lecture [258] that the link between trust and government had already been established by Confucius, whom she quotes with *"Three things are needed for government: weapons, food, and trust. If a ruler can't hold on to all three, he should give up weapons first and the food next."*

[26] Hardin [155] notes that David Hume proposed designing government institutions so they would serve our interets even if they were staffed by villains and scoundrels.

[27] Tocqueville visited the United States in the early 19th century from France and was impressed by its vigorous civil society, which he believed formed the basis for a truly democratic society: *"Americans of all ages, all stations in life, and all types of disposition, are forever forming association. There are not only commercial and industrial associations in which all take part, but others of a thousand types—religious, moral, serious, futile, very general and very limited, immensely large and very minute. ...Nothing, in my view, deserves more attention than the intellectual and moral associations in America"* [231].

ton Post, Harvard University, and the Kaiser Family Foundation, found similar evidence for the loss of trust in American society:

> America is becoming a nation of suspicious strangers, and this mistrust of each other is a major reason Americans have lost confidence in the federal government and virtually every other major national institution. Every generation that has come of age since the 1950s has been more mistrusting of human nature, a transformation in the national outlook that has deeply corroded the nation's social and political life [267].

Putnam concluded in his book that restoring civic engagement in America *"would be eased by a palpable national crisis, like war or depression or natural disaster, but for better and for worse, America at the dawn of the new century faces no such galvanizing crisis"* [275]. Returning to his initial survey sample shortly after the September 11, 2001, attacks, Putnam did indeed find that 51 percent of his respondents expressed greater confidence in the federal government in 2001 than they had a year earlier [276]. Similar levels of increasing trust could be noted not only for government, but also neighbors, co-workers, even total strangers. However, trust toward Arab Americans was about 10 percent below the level expressed toward other ethnic minorities.[28] Yamagishi and Yamagishi [380] call this kind of trust "particularized trust," in which one cooperates only with his or her own kind and close friends, compared with "generalized trust," which extends trust to "outsiders" as well. It is only the latter kind of trust which can truly produce social capital [336].[29]

Greasing the wheels of social interactions is not the only benefit that is attributed to trust in the social sciences. Sociolocists like Niklas Luhmann see trust also as an essential component for our everyday lives, as it reduces the complexity of our everyday risk assessments: *"A complete absence of trust would prevent [one] even getting up in the morning"* [225]. Simply by stepping out of our door, one is exposed to

[28] As this question was explicitly added after the September 11 events, no comparable data from before the attacks was available.

[29] Putnam uses the terms *bonding* and *bridging* social captial to differentiate the two: *"Bonding capital is good for undergirding specific reciprocity and mobilizing solidarity – bridging networks, by contrast, are better for linkage to external assets and for information diffusion. Moreover, bridging social capital can generate broader identities and reciprocity, whereas bonding social capital bolsters our narrower selves. Bonding social capital constitutes a kind of sociological superglue, whereas bridging social capital provides a sociological WD-40"* [275] (WD-40 is the brand name for a well-known light lubricant for rubber, metal, wood, and plastic. See `www.wd40.com`).

a multitude of risks by fellow citizens, such as being run over, assaulted, or robbed. Apart from trusting the benevolence of others (interpersonal trust), one also puts organizational trust in the employees of buses and trains (to come on time, to stop at the individual stations), airlines (not to crash airplanes into your city), or nuclear power plants (to follow the safety procedures). Last not least, one puts similar organizational (or systemic) trust in the effectiveness of the police to find and arrest violators, and in the judiciary system to prosecute them.

According to Endreß [102], this view of trust as a *"basic principle of social order"* has received increased attention as society advances technologically. Modern society is characterized by an increased functional differentation (i.e., we rely on an increasing number of specialists to perform various functions for us, such as plumbers, masons, doctors, lawyers, or butchers), and thus raises the number of interactions between previously unknown actors. Without relying on trust, coordinated actions under such conditions of extensive anonymity would be nearly impossible [157]. Under these circumstances, trust allows actors to "ignore" the looming risks and contingencies, and to facilitate coordinated and predictable interactions. An increase in telecommunication and telecooperation reinforces this necessity, as it greatly increases the number of interactions with hitherto unknown actors.

Research in the social and political sciences thus questions both the practicality and usefulness of replacing interpersonal or systemic trust with stricter rules of oversight or tighter technical enforcement. While increased transparency is certainly useful, it should lead less to accountability through micro-management, but more to good governance and honesty [258]. A reliance on trust in areas such as personal privacy might seem overly naïve at first, but seems less peculiar given the large number of trusting assumptions one makes in the everyday, often concerning much more valuable assets than postal addresses, such as health or finances.

Economies of Trust

Research in economic theory has long seen trust as an important factor for any form of cooperation and economic exchange.[30] A popular exam-

[30] Seeing trust as a key in economic exchanges relies on the so-called *Rational Choice Theory* – the idea that all human action is fundamentally 'rational' in character and that people calculate the likely costs and benefits of any action before deciding what to do. Its application to social interaction is called *social exchange theory* [314].

ple is the so-called *prisoner's dilemma* (PD), a game theoretic puzzle that was devised in the early 1950s at Rand corporation, a US-american think tank, as part of their research into global nuclear strategy [201].

In its classical form, the prisoner's dilemma describes the situation of two criminals that have been arrested by police and are interrogated separately. Each is offered a deal: if they confess to the crime while their accomplice remains silent, their testimony is used to ensure that the accomplice is receiving a substantial jail sentence. If both confess, each gets a jail sentence but receives a chance for an early parole. However, if both remain silent, the police will only be able to book them for some minor charges and will need to release them again in a few days.

Assuming that each prisoner, or player, is trying to maximise his own "payoff," without concern for the well-being of the other, the optimal strategy for each prisoner is to confess, even if both agreed beforehand to remain silent: Expecting his partner to confess would require one to make a confession a well, in order to minimize jail time. On the other hand, expecting that the partner remains silent would still prompt one to confess, as it would mean immediate release instead of doing (short) jail time for minor charges. Confessing is thus what is called the *dominant strategy* for both players, even though when both confess, it yields a lengthy jail sentence for both (this is the core of the dilemma).

This optimal outcomes changes, however, once it becomes possible to punish the other player for defecting. The *iterated* form of the prisoner's dilemma, for example, allows for punishing cheaters by cheating on them in turn in successive rounds. As Axelrod showed in 1984 [20], repeating such encounters over a long period of time with many players, each with different strategies, "greedy" strategies tend to do very poorly in the long run while more "altruistic" strategies do better, as judged purely by self-interest. When Axelrod invited academic colleagues all over the world to devise computer strategies to compete in an iterated prisoner's dilemma tournament, it turned out that the best deterministic strategy was "Tit for Tat."[31] The program would always cooperate on the first move and mimick the opponents previous move afterwards [360].

An even better strategy than simply mimicking the opponents behavior, however, is the "generous Tit for Tat" strategy, which also begins with a cooperative move and also repeats the opponents previous moves, but which will "throw in" a cooperative move after a series of

[31] "Tit for Tat" was also the simplest program entered, with only four lines of BASIC [360].

mutual defections, in order to see if it will in turn trigger cooperation on the part of the other player [230]. Such unfounded trust – sometimes called "optimistic" trust [155], as it overestimates the probabilities of trustworthiness[32] – can thus greatly benefit social and economic exchanges [69]. Consequently, social exchange theory assumes that trust emerges through the repeated exchange of benefits between two individuals [51].

This notion of trust as an economic enabler has since been validated in a number of real-world settings [41]. Bruhn [51] reports of studies that link strong elements of trust in a corporate culture to significant economic benefits. Fukuyama [127], examining the economic principles of a wide range of national cultures (Japan, China, Korea, Germany, France, and the United States), finds the same economic advantages on a macro-economic scale:

> It is no accident that the United States, Japan, and Germany were the first countries to develop large, modern, rationally organized, professionally managed corporations. Each of these cultures had certain characteristics that allowed business organizations to move beyond the family rather rapidly and to create a variety of new, voluntary social groups that were not based on kinship. They were able to do so because in each of these societies there was a high degree of trust between individuals who were not related to one another, and hence a solid basis for social capital [127].

Replacing trust with tools of bureaucracy, control, and surveillance not only misses out chances of cooperation, but also increases the overall production costs by creating and maintaining these often complex trust-replacement mechanisms. Obviously, blindly trusting in the face of untrustworthy behavior is similarly uneconomically. Instead, philosopher Onora O'Neill calls for *intelligent accountability*:

> [C]urrently fashionable methods of accountability damage rather than repair trust. If we want greater accountability without damaging professional performance we need *intelligent accountability*. ... Intelligent accountability, I suspect,

[32] Mansbridge [230] further differentiates between this optimistic trust, which overestimates trustworthiness for various nonmoral reasons, and *altruistic trust*, which is explicitly based on moral causes such as respect for the other. The latter not only benefits economic exchanges, but also improves society in general, as outlined in the previous section.

> requires more attention to good governance and fewer fantasies about total control [258].

In the context of privacy, a lack of sufficient trust thus seems to directly impact economic development in three ways:

1. *Merchant trustworthiness:* Service providers and on-line merchants have a high incentive to appear trustworthy to the consumer, as a lack of trust into a company can have a direct impact on its business performance: *"The battle is not for eyeballs; it's a battle for trust, hearts, and minds"* [243].

2. *Consumer trust disposition:* Similarly, consumers who do not trust either merchants, service providers, or the infrastructure that enables e-commerce with their personal data, potentially miss out on valuable opportunities.

3. *Enterprise management:* Trust plays also an important role within an enterprise, as customer data needs to be secured against both outside intruders and internal neglicence, while at the same time placing a sufficient amount of trust in its employees in order to establish a strong corporate culture that facilitates efficient business.

As with the previous two areas, psychological and social trust, it seems that economic incentives make trust part of any technical privacy solution, allowing merchants to become more trustworthy by supporting their customer data management, and letting consumers put more faith in on-line transactions through better transparency. Yet it also shows the limits of what technology alone can achieve, when excessive accountability tools threaten both efficiency and mutual respect within organisations.

3.1.3 Summary

This section has tried to explore the mechanisms behind trust and ethics, in order to assess their relevance to the problem of privacy protection and their usefulness as a complementary tool to any technical solution.

As we have seen, trust is both useful and essential. Trust is something that we cannot live without, as it makes it possible for us to live in a world of continuing uncertainties [225], realize more of our economic

potential through collaboration with others [129], and build large and efficient organizations [51]. And trust is something that we probably should not *want* to live without, as it is one of the most essential ingredients of stable societies and healthy democracies [344].

Privacy needs trust, just as any trade or exchange where goods do not synchronously change hands, like at a market, but where promises are made that one party will follow up on its duties later. While technology can and must support such promises, giving the benefit of doubt remains an important ethical aspect of such exchanges: Replacing trust through rigorous oversight (be it organizational or technological) creates a culture of mistrust [258] that cancels many of the benefits that a trusting society with its social captial offers [275]. So even if the intricacies of interpersonal, institutional, and systemic trust *could* be reliably and efficiently modelled through technology (which, given its elusiveness and ex-post nature, seems rather unlikely) – providing tools for promoting transparency instead of creating suspicion might in the long run help us to create both stable societies and more prosperous economies [127].

Our cultural norms and ethics provide the basis for such a trustful society, as trust is something that can neither be taught nor willed [226]. Norms and ethics permeate our daily lives, and create both formal and informal rules (through laws and social standards, respectively) that guide and limit our possible actions. By incorporating such social mechanisms into our design for privacy, we can hope for both a reduction in complexity, as well as an increase in efficiency. The next section will focus on the existing *formal* rules, the legal frameworks that form the other cornerstone of our integrative solution.

3.2 Legal Mechanisms

Relying on trust and social norms alone to guarantee one's privacy protection is risky, as the trustor's expectations about the future behavior of the trustee may turn out to be wrong. Laws are one of the most effective remedies against such inherent risks: *"Legal arrangements which lend special assurance to particular expectations and make them sanctionable ... lessen the risk of conferring trust"* [225]. Laws can thus often provide a substantial reduction in the risk of encountering a defaulting trustee, not only because they allow sanctions against cheaters, but more importantly because the threat of such sanctions can serve as

a "background structure" that deters actors from considering it in the first place [203].

More than hundred years after Warren and Brandeis laid the foundation for modern data protection laws, two distinctive principles for legal privacy protection have emerged: The European approach of favoring comprehensive, all-encompassing data protection legislation that governs both the private and the public sector, and the sectoral approach popular in the US that favors voluntary industry regulations whenever possible, employing legal constraints only when absolutely necessary.

The rise of the Internet and its World Wide Web in the early 1990s had prompted many to proclaim the demise of national legal frameworks, as their enforcement in a borderless cyberspace seemed difficult at least.[33] However, the opposite effect could be observed: At the beginning of the 21st century, many national privacy laws have not only been adjusted to the technical realities of the Internet, but also received a substantial international harmonization, thus facilitating cross-border enforcement.

3.2.1 Modern Privacy Laws

The first modern privacy laws that specifically addressed the rise of computerized, automated data processing (i.e., information privacy) were enacted in the 1970s in Europe, where the Second World War had taught many citizens the value of privacy: Many Jews in both Germany and the occupied territories had been identified by the Nazis through the comprehensive and detailed town registers that listed the religious orientation of their citizens freely [119]. Having just experienced the drawbacks of an "efficient" administration, strong sentiments resurfaced as European governments increasingly started to employ centralized data processing systems. The first data protection law in the world was enacted in the German state of Hesse in 1970, followed by similar laws and statues in Sweden (1973), the German state of Rhineland-Palatine, Austria (both 1974) and Germany (1977) [235]. The in Europe still popular term "data-protection" is a reminder of these very much technology oriented early privacy laws, which focused much more on the actual processing steps in computerized databases than trying to define

[33] In his 1996 "Declaration of Independence of Cyverspace," John Barlow declared *"Governments of the Industrial World, you weary giants of flesh and steel, I come from Cyberspace, the new home of Mind. On behalf of the future, I ask you of the past to leave us alone. You are not welcome among us. You have no sovereignty where we gather"* [25].

the privacy rights of the individual.

US Privacy Laws

While European data protection laws often specified regulations independently of the actual data collector, and thus applied both to governmental organizations as well as private enterprises, the US *Privacy Act of 1974* exclusively governed the data processing at the federal level [140]. This focus on regulating only governmental data processing is an important aspect of US privacy legislation, where the right to privacy is primarily anchored in the Fourth and Fifth Amendments[34] [322].[35]

- *Forth Amendment:* The right of the people to be secure in their persons, houses, papers, and effects, against unreasonable searches and seizures, shall not be violated, and no Warrants shall issue, but upon probable cause, supported by Oath or affirmation, and particularly describing the place to be searched, and the persons or things to be seized.

- *Fifth Amendment:* No person shall be ... compelled in any criminal case to be a witness against himself, nor be deprived of life, liberty, or property, without due process of law; nor shall private property be taken for public use, without just compensation.

By drawing support for privacy laws from the constitution, US jurisprudence lacks the means to extend these to private entities, as the constitution only describes the rights of citizens in relationship to their government, not to other citizens or companies[36] [57].

[34] The first ten amendments to the US Constitution have collectively become known as the "Bill of Rights." They were added as a result of objections to the original Constitution of 1787 during state ratification debates. Congress approved these amendments as a block of twelve in September 1789, and the legislatures of enough states had ratified ten of those twelve by December 1791 [358].

[35] In a landmark case, *Griswold vs. Connecticut* 1965, the US Supreme Court first explicitly recognized a *constitutional right to privacy*, drawing from the First, Third, Fourth, Fifth, and Ninth Amendments. The First Amendment guarantees freedom of worship, speech, and press. The Third provides that troops may not be quartered, (i.e., allowed to reside) in private homes without the owner's consent. The Ninth declares that the listing of individual rights is not meant to be comprehensive, i.e., that the people have other rights not specifically mentioned in the Constitution [358]. The case involved the directors of the Planned Parenthood League of Connecticut, a nonprofit agency which disseminated birth control information, who challenged a Connecticut law criminalizing contraceptives and counseling about contraceptives to married couples. The Court held that the law was unconstitutional, and specifically described two interests for protecting privacy: (1) "the individual interest in avoiding disclosure of personal matters" and (2) "the interest in independence in making certain kinds of important decisions" [322]. The latter is often referred to as decisional privacy, the former as informational privacy.

[36] An exception is the 13th Amendment, which prohibits slavery and thus also applies to private persons.

Up to today no comprehensive legal framework exists in the US that equally applies to governmental and private data collectors. It is left to industry associations to voluntarily enact self-regulations to respect the privacy of their customers. Only if specific problems emerge, individual sectoral laws are passed at the federal or state level. [322] lists some examples:

- *Fair Credit Reporting Act (FCRA) 1970*: Inspired by allegations of abuse and lack or responsiveness of credit agencies, US Congress passed the FCRA in 1970 to regulate credit reporting agencies. It requires credit reporting companies to provide individuals with access to their records, established procedures for correcting information, and sets limitations on disclosure.

- *Video Privacy Protection Act (VPPA) 1988*: When US President Ronald Reagan nominated the conservative Judge Robert Bork to the Supreme Court, the *Washington City Paper* checked up on his local video rental records, in the hopes of finding some not-so-conservative titles [305]. Incensed Congress quickly passed the VPPA, which has become known as the "Bork Bill," which generall prevents disclosure of personally identifiable rental records, such as titles of video cassettes rented or purchased, without the individual's written consent. While the Act might not often be invoked,[37] it is actually one of the strongest protections of consumer privacy against a specific form of data collection [110].

- *Driver's Privacy Protection Act (DPPA) 1994*: Selling motor vehicle records to private marketers had been common practice for US states for decades, including information such as one's name, address, phone number, Social Security number, medical information, height, weight, gender, eyecolor, photograph, and date of birth [322]. This practice ended only after the 1989 death of actress Rebecca Schaeffer, who was killed by an obsessed fan after he had obtained her home address through her motor vehicle record [107]. A series of similar murders and robberies, all planned on the basis of addresses obtained through motor vehicle records, quickly

[37] In 1997, an Oklahoma citizen complained that the academy award-winning German movie *The Tin Drum* contained child pornography and thus violated Oklahoma law. Police subsequently removed all copies of the movie from Oklahoma City video stores and obtained, without a warrant, the names of the people currently renting it. The list including a civil liberty activist who ended up sueing the City on the grounds of the VPPA (winning statutory damages of US$ 2500,-) [110].

prompted congress to pass the DPPA as an amendment the Violent Crime Control and Law Enforcement Act of 1994, limiting the release of personal information from a motor vehicle record to only governmental agencies.

- *Children's On-line Privacy Protection Act (COPPA) 1998*: The increased use of the Internet from children resulted in a sizable marketing industry catering specifically to lists with children's names on. After several investigative reports by national newspaper and television shows, which showed how easy it was for pedophiles to obtain a list of children and their ages for a specific geographic region, COPPA was enacted in 1998 and became effective in 2000 [106]. It protects the privacy of children under the age of 13 by requesting parental consent for the collection or use of any personal information of the users.

- *Health Insurance Portability & Accountability Act (HIPAA) 1996*: Congress enacted the HIPAA in 1996 in order to simplify the switching of health plans when changing jobs. However, as the act simplified data sharing, Congress was concerned about the resulting security and privacy issues of medical data. A set of regulations to address these issues was initially signed by the Clinton Administration at the end of its term in December 2000, but did not go into effect until 2002, as its implementation was delayed due to procedural errors and the significant changes made by the Bush Administration before its final enactment. The HIPAA regulations are the first comprehensive federal rules on health privacy [322].

- *Gramm-Leach-Bliley (GLB) Act 1999*: Just as the HIPAA, the primary purpose of the GLB Act was to 'modernize' an industry – in this case to facilitate the mergers of banks, brokerage companies, and insurance companies. And just as the HIPAA, the GLB Act's removal of red tape made it easier for such newly formed financial institutions to have access to large amounts of (previously separated) personal information, with no restrictions upon its use [108]. As a precaution, the GLB Act thus includes three requirements: the secure storage of personal data; advising customers of data sharing policies; and providing opt-out options to some of these sharings [322].

Note that for *civil* lawsuits – i.e., those between person (either humans or corporations), in contrast to the above *public* laws that govern disputes between the state and its citizens (again, either humans or corporations) – Prosser [273] documented four distinct privacy *torts* common in US law,[38] i.e., ways for an individual who felt his or her privacy had been violated to sue the violator for damages:

- *Intrusion* upon seclusion or solitude, or into private affiars;
- *Public disclosure* of emberrassing private facts;
- *Adverse publicity* which places a person in a false light in the public eye; and
- *Appropriation* of name of likeness.

Privacy torts are recognized by the individual US states (in contrast to the above sectoral laws, which apply on federal level),[39] though some states only recognize a subset of these torts.[40] Those opposed to enacting privacy regulation in the US on the federal level point out that these privacy torts recognized in most US state laws already provide an adequate level of protection [269].

Privacy as Property

One often discussed alternative to the enactment of strong, comprehensive privacy laws in the US is the "commodification" of personal information, i.e., treating personal data as *personal property* [218, 302][41]. Proponents of such a model suggest that this would allow individuals to better capture the value that their personal data has on the marketplace, while at the same time forcing companies to internalize the social cost currently borne by others through the widespread collection of use

[38] A *tort* is a civil wrong for which the law provides remedy [369]. The "law of torts" is part of the *common law*, which is the legal system of many anglo-american countries, such as the UK or the US. In contrast to *civil law* practiced in most European countries (which is derived from Roman law, and has the form of statutes and codes written and enacted by emperors, kings, and – today – by national legislatures), common law is based on traditions, customs, and precedents dating back to historical England [364].

[39] Some state, notably New York and Nebraska, do not recognize a common law basis for torts but instead provide statutory (i.e., written and enacted by the state government, instead judicially through interpretation of common law [368]) protection [322].

[40] Some states, such as Minnesota, North Dakota, and Wyoming, did not recognize any of those privacy torts until as recently as 1998 [322].

[41] See also Alessandro Acquisti's Web page on the economics of privacy at `www.heinz.cmu.edu/~acquisti/economics-privacy.htm`

of personal data,[42] thus implicitly prompting data collectors to "make better investment decisions about what data to collect and what uses to make of the data" [302]. In order to alleviate the possibly substantial transaction costs for individuals if they would have to negotiate separate sales agreements for each data exchange, intermediary businesses, sometimes dubbed "infomediaries" [151], would negotiate with buyers on behalf of data subjects, taking a small fee of the sales revenues for their service.

Maybe the most convincing argument for regarding privacy as property is that this might increase support for strong privacy laws in the US: *"If you could get people (in America, at this point in history) to see a certain resource as property, then you are 90 percent to your protective goal. If people see a resource as property, it will take a great deal of converting to convince them that companies . . . should be free to take it. . . . That would be 'theft,' and this is my point: 'theft' is positively un-American"* [218].

However, several problems exist with this seemingly simple and straightforward approach. As Samuelson points out, the most common justification for property rights is to enable markets to more efficiently allocate a scarce resource [302]. However, personal data seems to be anything but scarce – it is information privacy that is in short supply. Also, property rights are typically alienable, i.e., the buyer can freely transfer to a third party whatever was acquired from the seller. In comparison to used cars or land, sellers of personal data often care strongly about whom this data is passed on. Last not least, it remains far from obvious that simply by passing property laws such a functioning market would come into being. Chief among such concerns is the cost of a proper infrastructure to support such an information market, just as today's seller of intellectual property (e.g., movies or music) need to regulate the distribution of their assets with sophisticated digital rights management systems.[43] Samuelson instead suggests modeling "marketable" privacy laws along trade secrecy regulation, which is an established concept in US legislation and shares three important characteristics with such an envision data protection law: the seller's interest

[42] As Swire and Litan point out, companies do not *"suffer losses from the disclosure of private information . . . In economic terms, the company internalizes the gains from using the information but can externalize some of the losses and so has a systematic incentive to overuse it"* [332].

[43] Kenneth Laudon [211] envisions the need to assign every participant in such a "National Information Market" (NIM) a unique identifier, comprised of a set of public-key-cryptography key-pairs, which would help to keep track of data flows.

to restrict access to and unauthorized uses of the information; the interest of giving both parties control over a commercial exploitation of this information; and an interest in enforcing "minimum standards of commercial morality" [302].

EU Privacy Laws

On the other side of the Atlantic, a much more civil libertarian perspective on pesonal data protection prevails. Individual European states began harmonizing their national privacy laws as early as the mid-1970s. In 1973 and 1974, the European Council[44] passed resolutions (73)22 and (74)29, containing guidelines for national legislation concerning private and public databases, respectively [76, 77]. In 1985, the "Convention for the Protection of Individuals with regard to Automatic Processing of Personal Data" (108/81) went into force, providing a normative framework for national privacy protection laws of its member states [78]. However, even though 31 of its member states have signed the convention so far,[45] its effect on national laws has still been rather limited [235]. It was the 1995 "Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data" [94] (in the following simply called "the Directive") that achieved what Convention 108/81 set out to do, namely a lasting harmonization of the various European data protection laws and an effective international tool for personal privacy even across European borders.

The Directive has two important aspects that advance its international applicability. On the one hand, it requires all EU member states[46] to enact national law that provides at least the same level

[44] The European Council was founded in 1949 in order to harmonize legal and social practices across Europe. It groups together 45 countries – apart from the 25 EU member states mostly central and eastern European countries. Since 1989, its main job has become assisting the post-communist democracies in central and eastern Europe in carrying out political, legal and economic reform.

[45] As of August 2004. See `conventions.coe.int/Treaty/Commun/ChercheSig.asp?NT=108&CM=8&DF=8/18/04&CL=ENG` for latest figures.

[46] The directive actually applies to the so-called "European Economic Area" (EEA), which not only includes the EU-member states but also Norway, Iceland, and Liechtenstein. The agreement creating the EEA was negotiated between the EU and seven member countries of the EFTA and signed in May 1992. Subsequently one of these (Switzerland) decided after a referendum not to participate, and three others (Austria, Sweden, and Finland) joined the Union. The EEA Agreement entered into force on 1 January 1994. The EEA was maintained because of the wish of the three remaining – Norway, Iceland and Liechtenstein – to participate in the Single Market, while not assuming the full responsibilities of membership of the EU.

of protection as the Directive stipulates.[47] This European harmonization allows for a free flow of information among all its member states, as personal data enjoys the same minimum level of protection set forth by the Directive.

On the other hand, its article 25 explicitly prohibits the transfer of personal data into "unsafe third countries," i.e., countries with data protection laws that do not offer an adequate level of protection as required by the Directive. After European officials made it clear that they intended to pursue legal action against the European branch offices of corporations that would transfer personal data of EU-citizens to their corresponding headquarters in such unsafe third countries, a large number of non-European countries around the world began to adjust their privacy laws in order to become a "safe" country with regards to the Directive, and thus become part of the European Internal Information Market.[48]

The Safe Harbor Agreement

The Directive had also a direct impact on US legislation. From a EU point of view, the sectoral approach in the US does not provide sufficient protection for the personal data of EU citizens, which would – according to article 25 of the Directive – require companies to cease transatlantic personal data transfers into the US. After years of negotiation, both sides agreed on a compromise in July 2000: In accordance with article 26 of the Directive, which provides for exceptions on the basis of explicit contractual clauses, the EU allowed US companies to voluntarily declare their adherence to the principles of the Directive and subsequently be exempt from the transfer ban. This arrangement, called *Safe Harbor Agreement*,[49] had been signed by over 550 companies by August 2004, including companies such as Amazon, DoubleClick, General Motors, Hewlett Packard, Intel, IBM, Merck, Oracle, and Procter & Gamble. However, many privacy advocates feel that Safe Harbor provides significantly less protection than EU privacy laws, for example when it comes to the right to inspection and correction of stored

[47] All 15 pre-2004 member states have enacted national laws compatible to the Directive by now. Of the ten countries that joined the EU on May 1st, 2004, many have not yet updated their legislation to full compliance. However, citizens in these countries can already take an issue to their national courts based on the Directive, even if no national legislation exists yet.

[48] As of August 2004, Argentina, Canada, Switzerland, and the British Channel Islands Guernsey and Isle of Man were considered "safe" third countries with respect to personal data transfers.

[49] See `www.export.gov/safeharbor/sh_overview.html`

data, or regarding compensation for wrongfully processed data [284].

Despite the criticism, the Safe Harbor Principles – even if being only a weakened version of the principles of the Directive – do constitute a significant increase in privacy protection for private data collections in the US, as the announcement requirements of data collections, the provision of anonymous and pseudonymous access alternatives, and the needed correction mechanisms do exceed the minimum standards typically found in US companies' privacy regulations.

3.2.2 The Fair Information Practices

The minimum standards regarding the collection and procession of personal data that have been incorporated into the Directive have their roots in a 1973 report of the *United States Department for Health Education and Welfare* (HEW), which set forth a list of *Fair Information Practices* that have been a staple of privacy law not only in the US (especially the *Privacy Act* of 1974), but worldwide [270]. The five principles are as follows [321]:[50]

1. *Collection limitation.* There must be no personal data record keeping systems whose very existence is secret.

2. *Disclosure.* There must be a way for an individual to find out what information about him is in a record and how it is used.

3. *Secondary usage.* There must be a way for an individual to prevent information about him that was obtained for one purpose from being used or made available for other purposes without his consent.

4. *Record correction.* There must be a way for an individual to correct or amend a record of identifiable information about him.

5. *Security.* Any organization creating, maintaining, using, or disseminating records of identifiable personal data must assure the reliability of the data for their intended use and must take precautions to prevent misuse of the data.

In the early 1980s, the Organization for Economic Cooperation and Development (OECD) took up those principles and issued "The OECD

[50] The original report is available from `aspe.os.dhhs.gov/datacncl/1973privacy/tocprefacemembers.htm`

Guidelines on the Protection of Privacy and Transborder Flows of Personal Data" [260], which described eight practical measures aimed at harmonizing the processing of personal data in its member countries. By setting out core principles, the organization hoped to *"obviate unnecessary restrictions to transborder data flows, both on and off line."* The eight principles are as follows:[51]

1. *Collection Limitation Principle.* There should be limits to the collection of personal data and any such data should be obtained by lawful and fair means and, where appropriate, with the knowledge or consent of the data subject.

2. *Data Quality Principle.* Personal data should be relevant to the purposes for which they are to be used, and, to the extent necessary for those purposes, should be accurate, complete and kept up-to-date.

3. *Purpose Specification Principle.* The purposes for which personal data are collected should be specified not later than at the time of data collection and the subsequent use limited to the fulfillment of those purposes or such others as are not incompatible with those purposes and as are specified on each occasion of change of purpose.

4. *Use Limitation Principle.* Personal data should not be disclosed, made available or otherwise used for purposes other than those specified in accordance with the Purpose Specification principle except:

   a) with the consent of the data subject; or

   b) by the authority of law.

5. *Security Safeguards Principle.* Personal data should be protected by reasonable security safeguards against such risks as loss or unauthorized access, destruction, use, modification or disclosure of data.

6. *Openness Principle.* There should be a general policy of openness about developments, practices and policies with respect to personal data. Means should be readily available of establishing the existence and nature of personal data, and the main purposes of

[51] These principles are reprinted from `www.junkbusters.com/ht/en/fip.html`

their use, as well as the identity about usual residence of the data controller.

7. *Individual Participation Principle.* An individual should have the right:

   a) to obtain from a data controller, or otherwise, confirmation of whether or not the data controller has data relating to him;

   b) to have communicated to him, data relating to him

      i. within a reasonable time;

      ii. at a charge, if any, that is not excessive;

      iii. in a reasonable manner; and

      iv. in a form that is readily intelligible to him;

   c) to be given reasons if a request made under subparagraphs (a) and (b) is denied, and to be able to challenge such denial; and

   d) to challenge data relating to him and, if the challenge is successful, to have the data erased; rectified, completed or amended.

8. *Accountability Principle.* A data controller should be accountable for complying with measures which give effect to the principles stated above.

Even though the OECD principles, just as the HEW guidelines before them, carried no legal obligation, they nevertheless constituted an important international consensus that substantially influenced national privacy legislation in the years to come [322].

Taken together, the two sets of guidelines above are often summarized in five basic principles: Openness; data access and control; data security; data minimization; and individual consent. Especially the last point – getting the consent of the data subject before the data collection – has received increased attention in the last years. Even though already the rather technically oriented privacy laws of the 1970s stipulated the possibility for the individual to correct his or her stored data, this was more in the spirit of ensuring the accuracy of the stored data, rather than questioning the legitimacy of the collection [235].

It took until the 1980s before revised European laws began to view privacy more and more as an individual right that people should be able to exercise without unnecessary burden. Representative for this

paradigm shift was the so-called "census-verdict" of the German federal constitutional court (Bundesverfassungsgericht) in 1983, which extended the existing *right to privacy of the individual* (Persönlichkeitsrecht)[52] with the *right of self-determination over personal data* (informationelle Selbstbestimmung)[53] [235].[54]

The judgment reads as follows:[55]

> If one cannot with sufficient surety be aware of the personal information about him that is known in certain part of his social environment, ... can be seriously inhibited in his freedom of self-determined planning and deciding. A society in which the individual citizen would not be able to find out who knows what when about them, would not be reconcilable with the right of self-determination over personal data. Those who are unsure if differing attitudes and actions are ubiquitously noted and permanently stored, processed, or distributed, will try not to stand out with their behavior. ... This would not only limit the chances for individual development, but also affect public welfare, since self-determination is an essential requirement for a democratic society that is built on the participatory powers of its citizens [285].

The then president of the federal constitutional court, Ernst Benda, summarized his private thoughts regarding their decision as follows:[56]

> The problem is the possibility of technology taking on a life of its own, so that the actuality and inevitability of technology creates a dictatorship. Not a dictatorship of people over people with the help of technology, but a dictatorship of technology over people [285].

The concept of self-determination over personal data[57] constitutes an important part of modern privacy legislation with respect to ensuring

[52] See `www.eurofound.eu.int/emire/GERMANY/RIGHTTOPRIVACYOFTHEINDIVIDUAL-DE.html`

[53] See `www.eurofound.eu.int/emire/GERMANY/RIGHTOFSELFDETERMINATIONOVERPERSONALDATA-DE.html`

[54] The finding was triggered by the controversy surrounding the national census announcement on April 27, 1983, which chose the unfortunate wording "Totalzählung" and thus resulted in more than hundred constitutional appeals (Verfassungsbeschwerde) to the federal constitutional court [285].

[55] Translation by the author.

[56] Translation by the author.

[57] Often abbreviated to *data self-determination*.

the autonomy of the individual. Firstly, it extends the fair information principles with a participatory approach, which would allow the individual to decide beyond a "take it or leave it" choice over the collection and use of his or her personal information. Secondly, it frames privacy protection no longer only as an individual right, but emphasizes its positive societal role. Privacy not as an individual fancy, but as an obligation of a democratic society, as Julie Cohen notes:

> Prevailing market-based approaches to data privacy policy ... treat preferences for informational privacy as a matter of individual taste, entitled to no more (and often much less) weight than preferences for black shoes over brown, or red wine over white. But the values of informational privacy are far more fundamental. A degree of freedom from scrutiny and categorization by others promotes important non-instrumental values, and serves vital individual and collective ends [68].

Modern European privacy laws that conform to the Directive additionally provide for a number of protection mechanisms that are designed to strengthen the usually weak bargaining position of the individual. Article 8 of the Directive provides a blanket protection against processing sensitive information such as ethnicity, religious beliefs, political or philosophical views, union membership, sexual orientation, and health, unless for medical reasons or with the explicit consent of the data subject [235].

3.2.3 Law Enforcement Issues

In the immediate aftermath of the September 11, 2001, attacks on the New York World Trade Center towers, many national governments and international bodies enacted a range of laws and regulations with the aim to strengthen national security and suppress terrorism. According to the 2003 EPIC Privacy and Human Rights Report [212], four trends may be identified:

1. *Increased Communications Surveillance and Search and Seizure Powers:* Many national initiatives significantly reduced the authorization and oversight requirements for wiretapping and searches. In addition, existing laws were often updated to increase the breadth

of application of these powers to include novel technology and communication infrastructures.

2. *Weakening of Data Protection Regimes:* Existing requirements for mandatory destruction of personal data after its purpose had been fulfilled (e.g., call records for billing purposes) have often been lifted in order to permit *"the retention of critical data for a reasonable period"* [212]. Similarly, information access rights have often been reduced in the interest of national security and infrastructure protection, thus limiting government accountability.

3. *Increased Data Sharing:* Several policies were introduced to enable and promote data sharing both within and across government agencies, as well as between government and private sector data collectors.

4. *Increased Profiling and Identification:* The most immediate activity since September 2001 has been the extensive profiling of air travelers. Also, many governments have been updating national identification schemes for citizens and non-citizens, e.g., by including biometrics in national ID cards and increasing border controls for non-citizens. Even countries with a well-known disposition against national ID-cards, such as the UK and the US, have repeatedly considered introducing such schemes.

While privacy protection had always to strike a balance between individual liberty and public safety (cf. section 2.1.3 above), the threat of terrorism has significantly altered the scales that are used to measure the pros and cons of personal privacy. Four examples of recently introduced legislation illustrate the above trends: the USA PATRIOT Act, the Terrorist Information Awareness project, the EU Telecommunications Directive, and the UK Terrorism Act.

The USA PATRIOT Act

Surveillance of wire, oral, and electronic communications for criminal investigations in the US is governed by the Omnibus Safe Streets and Crime Control Act of 1968 and the Electronic Communications Privacy Act of 1986 ("Title III"). It requires police to obtain a court order based on several legal requirements before it can begin capturing the content of a communication.

In the wake of the September 11, 2001 attacks, US Congress passed the USA PATRIOT Act,[58] which substantially lowers the requirement for conducting wiretaps, both for traditional areas such as telephone surveillance, as well as for electronic communication. It allows prosecutors to certify that a certain surveillance action would collect information relevant to an ongoing investigation, rather than having to obtain a full-fledged warrant, which involves substantially more prior evidence [100].[59] Judges have no jurisdiction to reject such a certification, practically granting investigators free reign over surveillance activities as long as these "help to defend terrorism" [223]. It also simplifies surveillance operations across a wide variety of technologies by significantly broadening existing definitions.

Several immigrant tracking programs are also part of PATRIOT. The US VISIT program[60] requires visitors to submit a biometric identifier to the government upon entry. Immigration authorities have also begun implementing SEVIS,[61] an Internet-based system that requires schools to transmit student information such as their identifaction, academic data, and disciplinary information, to Immigration Services for the duration of the student's stay in the US [212].

The Terrorism Information Awareness Project

The Terrorism Information Awareness (TIA) project, or "Total Information Awareness", as it was initially called, is a program of the Defense Advanced Research Projects Agency (DARPA), the central research and development organization of the US Department of Defense. It is very similar to the CAPPS-II [62] system for airline profiling, as it tries to detect the "information signature" of terrorists by scanning databases of personal information. However, TIA was conceived with a much greater scale in mind than CAPPS-II, eventually being able to cover *all* available databases in the US, both governmental and private, as well as any foreign databases that would be made available by the respective governments, in its search for *"terrorists and criminals involved*

[58] "USA PATRIOT" is an acronym for "Uniting and Strengthening America by Providing Appropriate Tools Required to Intercept and Obstruct Terrorism."

[59] This means that the person whose communications are subject to this order need not be a criminal suspect at all; all that is required is a certification that this information is relevant to an investigation [322].

[60] "US VISIT" is an acronym for "United States Visitor and Immigrant Status Indicator Technology."

[61] "SEVIS" stands for "Student and Exchange Visitor Information System."

[62] Enhanced Computer-Assisted Passenger Profiling System

in 'low-intensity/low-density' forms of warfare and crime" [109].

Funding for TIA has officially been cut by US congress in September 2003 [109], as has support for CAPPS-II [152]. However, many of its former subprograms are continued and similar programs are newly launched, such as the "Matrix" program, which aims to give state law enforcement agencies new tools to find patterns in both private and public databases, e.g., allowing investigators to *"instantly find the name and address of every brown-haired owner of a red Ford pickup truck in a 20-mile radius of a suspicious event"* [256].

The EU Electronic Communications and Privacy Directive

While the Data Protection Directive 95/46/EC placed severe restrictions on the retention of collected data (see page 72), the new EU Electronic Communicatios and Privacy Directive 2002/58/EC (often called the "e-Privacy Directive") that passed in May 2002 reversed this requirement, allowing each EU member state to pass legislation to retain traffic and location data of mobile phones, SMS, landline telephones, faxes, e-mails, chatrooms, or any oher electronic communication devices [212].

The 2002 e-Privacy Directive updates the 1997 Telecommunications Privacy Directive, which in turn particularised and complemented the 1995 Directive 46/EC for use in the telecommunications sector [96]. The 2002 update was thought necessary to take technological changes into account and to make the provisions as technology-neutral as possible. Among other things, it for example regulates the use of unsolicited e-mail in Europe, enforcing a strict "opt-in" requirement. However, its most influential effect on EU privacy legislation lies in the new exceptions for data retention granted not only for the purpose of national security, but generally for any criminal investigation, as well as to both prevent and prosecute criminal offenses, all without specific judicial authorization [212].

A number of European countries have already established new data retention laws in line with the e-Privacy Directive (e.g., Belgium, Denmark, Fance, Spain, Switzerland and the UK), while others such as Austria, Germany, or Italy still question whether such retention would be compatible with the respective national laws [212].

The UK Regulation of Investigatory Powers Act

The United Kingdom is one of the countries with data retention legislation already in place. The interception of communications is regulated in the UK by the Regulation of Investigatory Powers Act (RIPA) of 2000.[63] Part I authorizes any public authority designated by the Home Secretary[64] to access any "communications data" without a warrant. It also requires communications service providers to provide "reasonable interception capabilities" in their systems [212].[65]

The power of the bill (and thus the controversity surrounding it) comes when seen in combination with the Anti-terrorism, Crime and Security Act (ATCSA) of 2001, which passed three months after the September 11 attacks.[66] It sets out a code of practice to communications provider to actually retain all kinds of communication data for the purpose of protecting national security or preventing or detecting crimes that relate to national security. However, as RIPA allows designated public authorities access to any stored communications data, no matter for what purpose these are stored, the combination of ATCSA and RIPA effectively discloses personal communication information to any such authority for the total duration of the (national-security related) retention period – which currently is proposed to be seven years [212] – for reasons that have no connection (direct or indirect) with national security [174].

3.2.4 Summary

The balance between anonymity and responsibility, between privacy and security, as defined in today's data protection laws around the world, is not an absolute, but a fragile interplay that must constantly be re-examined, depending upon technical possibilities and social needs. Particularly in times of technological change, where circumstances that were not yet foreseeable at the time of the law's conception substantially alter the playing field, this reinterpretation, restating, and reformulation cannot be avoided.

Whether recent developments herald a coming age of governmental surveillance and powerful police states, or if in fact today's privacy

[63] Available at `www.homeoffice.gov.uk/crimpol/crimreduc/regulation/`

[64] The Home Secretary is the minister responsible for law and order in England and Wales [367].

[65] Part II of the RIPA covers the use of covert surveillance, agents, informants and undercover officers, while part III covers the investigation of electronic data protected by encryption.

[66] Available at `www.legislation.hmso.gov.uk/acts/acts2001/20010024.htm`

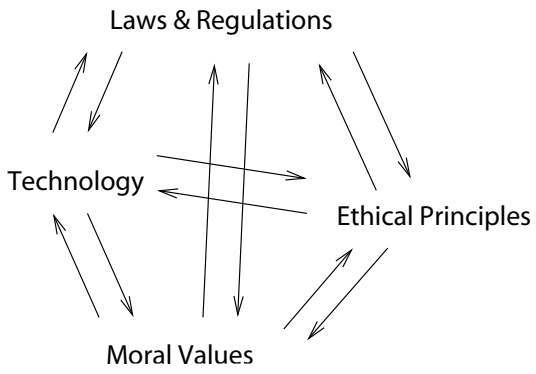


Figure 3.3: *Complex Interrelations.* Moral beliefs, ethical principles, laws, and technology, all influence each other, creating a complex web of interactions where changing one aspect always has an effect on the others. New technology could thus significantly alter our ethical principles, for example when memory amplifiers would allow comprehensive recordings of all private conversations.

legislation provides us with an unprecedented level of protection never before enjoyed in history – technical solutions cannot operate independently of both public morals and society's norms and laws. While technology shapes what kind of laws can be implemented, so do legal realities influence what kind of technology can be deployed.

Figure 3.3 illustrates this intricate web of dependencies between these three areas. The last two sections have tried to provide a comprehensive review of the boundaries set by social and legal mechanisms, within which we will need to position our technical solutions. For example, while we might be able to construct technology that completely anonymizes two or more parties that electronically interact with each other, legal frameworks set forth to protect society from crimes would prevent us from fielding it. Similarly, while we might not be able to certify the trustworthiness of a (known or unknown) service provider, we might instead use technology to provide cues that humans can establish trust upon, and laws to create strong incentives for service providers to act trustworthy upon this trust.

Our brief ventures into the social and legal realms might not have resulted in exact specifications for our final prototype. However, having the "big picture" in mind during our design phase should lessen our risks of providing only shallow and short-lived remedies – either because they are incompatible with our social realities, or inconsistent with our existing legal frameworks. Now all that is missing before we can assemble our list of guiding principles is a brief view at the technically

possible – our toolbox of algorithms and systems that we can readily assemble bits and pieces of our infrastructure from. The next section will thus look at some very basic and well-known technology, mainly from the area of Internet privacy and security, that we will build upon in chapter 4 when introducing PawS, our privacy-awareness system.

3.3 Technical Mechanisms

Technical tools form the last building block in our "Privacy Mechanisms Toolbox", though in contrast to the previously described social and legal mechanisms they are much more readily deployable in our architecture.

The following sections will provide a quick overview of the available systems and mechanisms in three areas: encryption and authentication tools that allow us to keep communication between two parties private and that support access control to stored information; anonymity and pseudonymity tools that facilitate anonymous access whenever identification is not necessary, or that provide the means of using a fixed pseudonym for repeated interactions; and transparency and trust tools that can be used to provide background information about data collections and the data collectors behind it, allowing data subjects to better judge for themselves whether they want to disclose any personal information.

Note that the selection is quite specific to our system, i.e., it focuses on those mechanisms that are part of our PawS architecture, even though many other systems and mechanisms potentially exist.

3.3.1 Encryption and Authentication Tools

When exchanging information with a service provider, e.g., uploading personal information in order to subscribe to a certain service, agreements with the data collector are useless if an unknown third party can easily listen in and use this information for its own goals. Encryption tools will allow us to *prevent others from eavesdropping* in on our information exchange with a data collector.

Similarly, once personal information has been transmitted to a data collector, the storage of such data must be made reasonably secure, allowing only authorized persons to access this information. Authentication mechanisms can make sure that only authorized persons, i.e., who posess correct credentials, can read, modify, or delete such data.

PawS makes use of two established technologies to secure both the communication between data subjects and data collectors (as well as within PawS itself) as well as access to stored information: a cryptohraphic protocol called SSL (Secure Sockets Layer) [126] for secure communication and endpoint authentication, and the XML-DSIG standard [27], an implementation of *digital signatures* for XML. Both employ *public key cryptography*, a concept that allows key exchange to happen over unsecured connections (i.e., "in plain view" of any potential attacker) without compromising the security of the encryption process. The following section will briefly summarize the basic concept of public key cryptography before outlining the SSL protocol as well as the XML digital signature standard.

Public Key Cryptography

Conventional cryptosystems rely on the fact that both the encyphering and the decyphering end of a connection use the same key. This means that this key needs to be kept absolutely secret and that both sender and receiver of such an encrypted message must agree upon a particular key before they begin exchanging messages, e.g., by meeting in person or using a trusted courier.

With public key cryptography, this "key distribution problem" is solved, as it allows two parties to agree on a common key over an insecure channel without having to exchange that key beforehand. The basic idea is to use a pair of *two* keys – one private (which must be kept absolutely secret), the other public (which can be widely distributed). Due to the mathematical nature of creating a key pair, deducing one from the other should not be possible. The idea was first proposed by Diffie and Hellman in their 1976 paper "New Directions in Cryptography" [91] and incorporated by Rivest et al. into a practical algorithm – called the *RSA cipher* – in 1978 [294].

One important feature of such a keypair is that both keys can be used interchangeably, i.e., data encrypted with the public key can only be decrypted using the private key, but data encrypted with the private key can only be decrypted using the matching public one. This allows this general mechanism to not only support standard data encryption by facilitating key exchange, but also the concept of signed messages that can be used to authenticate the sender of a particular message.

Note that the security of a cryptographic system based on public key exchange relies on more factors than just the algorithm (e.g., RSA)

and its implementation. Besides the need for keeping the private key absolutely private, the correct attribution of a public key to its owner requires either a meeting in person or a trusted courier (just as conventional single-key cryptography), or a *public key infrastructure (PKI)* in which an operator (the certificate authority) can attest that a certain public key actually belongs to a certain person. An alternative are 'open' PKIs in which anyone can attest the authenticity of someone else's public key, lowering infrastructure costs but also increasing the possibility of falsely attributed identities [361].

Compared to conventional, symmetric cryptosystems, public key cryptography is more complicated and implementations thus typically run much slower. In practice, public-key-based systems (also called asymmetric cryptosystems) are therefore often used to only facilitate the secret exchange of a common key, which can then be used with a fast symmetric encryption method.

A good introduction into public key cryptography and its applications can be found in [309].

SSL and TSL

Secure Sockets Layer (SSL) [126] and Transport Layer Security (TLS) [90], its successor, are cryptographic protocols that provide secure communications on the Internet.[67] They are most often used to secure an HTTP channel (then called an "HTTPS" connection) but can also be used with, e.g., SMTP to secure mail delivery, as they run on layers just above the TCP transport protocol [362].

SSL involves three basic steps in order to secure a communication connection:

1. Negotiate supported protocol levels.

2. Exchange encryption key and authenticate server[68] using public key cryptography and certificates.

3. Agree on a shared symmetric key to encrypt bulk traffic data.

[67] We will use the term SSL in the following to mean both SSL and TLS, unless otherwise noted. In practice, most systems in use today support TLS while being able to transparently downgrade a connection to SSL 3.0.

[68] While SSL could in theory authenticate both parties in a secret communication, the lack of available public key infrastructures mean that typically only one party – the server – is authenticated using a certificate signed by a trusted certificate authority.

Since SSL is a modular protocol, it can support a number of different algorithms to do the actual key exchange and data encryption. The corresponding capabilities of both the server and the client need therefore to be matched first before any encryption can take place. After this step, the server sends out a server certificate (containing its public key) that has been signed by a certificate authority. This allows the client to verify the identity of the server. The client then creates a random key (called the *pre-master secret*) that it sends back to the server after encrypting it with the server's public key. Both client and server use this pre-master secret to compute a *master secret*, which in turn is then used to compute a shared *session key*. This session key then constitutes a symmetric key that allows client and server to exchange messages using any symmetric cryptographic protocol.[69]

SSL thus provides the following security services to all upper protocol levels:

- *Confidentiality* is achieved by encrypting all data transmissions.
- *Server authentication* (and optionally client authentication) is provided through the use of certificates.
- *Data integrity* if possible through the use of one-time random numbers (*Nonce*) in transmissions, effectively preventing replay attacks.

Digital Signatures and XML-Signature

Using a user's public key to encrypt certain data can insure that only the user (using her matching private key) can decrypt this information. However, due to the interchangeable nature of the keys in a keypair, we can reverse this process, allowing a user to encrypt information using her private key, which can only be decrypted using the user's public key. While seemingly useless from a secrecy point of view (as anybody could be in possetion of the user's public key – it is public, after all), this can be used to provide *authenticity* to messages: As only the user herself should be in posession of the secret key, having a message that can be decrypted using the user's public key proves that it was encrypted with the user's private key, thus implying the origin of the message.[70] These

[69] The symmetrical protocol typically uses DES, Triple DES, or the newer Rijndael/AES (see `http://csrc.nist.gov/CryptoToolkit/aes/`).

[70] This is why the private key must be kept absolutely private – otherwise this strong link between private key and the user's identity cannot be maintained.

```
<Signature Id="MyFirstSignature" xmlns="http://www.w3.org/2000/09/xmldsig#">
  <SignedInfo>
  <CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n"/>
  <SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#dsa-sha1"/>
  <Reference URI="http://www.w3.org/TR/2000/REC-xhtml1-20000126/">
    <Transforms>
      <Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n"/>
    </Transforms>
    <DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
    <DigestValue>j6lwx3rvEPO0vKtMup4NbeVu8nk=</DigestValue>
  </Reference>
</SignedInfo>
  <SignatureValue>MC0CFFrVLtRlk=...</SignatureValue>
  <KeyInfo>
    <KeyValue>
      <DSAKeyValue>
        <P>...</P><Q>...</Q><G>...</G><Y>...</Y>
      </DSAKeyValue>
    </KeyValue>
  </KeyInfo>
</Signature>
```

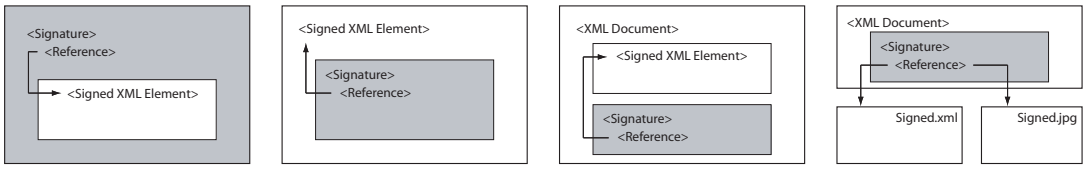
Figure 3.4: *Example of an XML-Signature*: The `SignedInfo` element contains the information (the document) that is being signed, with each signed object being referenced with a `Reference` element. The `SignatureValue` is the result of applying the `SignatureMethod` on a canonicalized version (using the `CanonicalizationMethod`) of the signed object [27].

two concepts can also be combined, allowing a user to first encrypt a message (or a part of it) with her private key and then using the recipients public key to encrypt it again, thus creating a message that only the intended recipient can read and only the alleged sender could have encrypted.

In practice, digital signature algorithms do not operate on an entire message but on a condensed version of it, the so-called *message digest*, which is computed from the orginal message using a *hash function*,[71] as hashing and then encrypting a short message digest is typically much faster than encrypting an entire message using public key cryptography.

Figure 3.4 shows an example of a digital signature using the XML-Signature syntax [27]. Using XML-signatures to sign messages allows not only a common message syntax in XML-based applications (such as PawS), but also facilitates XML signatures that are part of the XML document they are signing, thus achieving a high level of data encapsu-

[71] Hash functions take a long string as input and compute a fixed-length string as output such that a) it is hard to recompute the original string from it (called *preimage resistance*), b) it is hard to find a different input string that will yield the same hash output as the given input string (called *second preimage resistance*), and c) it is hard to find *any* two different input strings that result in the same output string (called *collision resistance*) [355].



(a) Enveloping (b) Enveloped (c) Detached (d) External

Figure 3.5: *XML-Signature formats.* XML Signatures can contain the signed element (a), can themselves be contained in the elements they sign (b), or can be completely separate from the signed element (c), even signing external documents or pictures (d) [296].

lation. The XML-Signature specification in fact defines three signature types: *enveloping* signatures contain the data they sign, *enveloped* signatures are contained within the data they sign, while *detached* signatures – such as the example in figure 3.4 – sign sibling elements or even external network resources [27]. A visual overview can be seen in figure 3.5. Since PawS uses XML to encode its messages, XML-signatures are ideally suited to provide message authenticity and tamper-protection.

It is important to note that the status of using digital signatures as evidence in a legal proceeding is still controversial, as – in contrast to handwritten signatures – digital signatures can be generated automatically without the "signer's" knowledge. While a digital signature can significantly increase the chances that an electronic communication has not been tampered with, it does not in itself guarantee that the message has been sent by the party signing it. However, many countries have already passed electronic signature laws that qualify digital signatures as legally binding, just as their handwritten counterparts (though some exceptions often apply), such as the EU's Directive 1999/93/EC on Electronic Signatures [95] or the US "Electronic Signatures in Global and National Commerce Act" [115].

3.3.2 Anonymity and Pseudonymity Tools

Most modern privacy legislation requires that whenever possible, *"anonymous or pseudonymous access . . . must be offered whenever technically possible"* [96]. [263] defines anonymity as "the state of being not identifiable within a set of subjects, the *anonymity set*." The "Common Criteria for IT Security Evaluations (CC)" standard, also known as ISO 15408, states that "[Anonymity] ensures that a user may use a resource or service without disclosing the user's identity" [178].

On the Internet, protecting one's anonymity is foremost a question

of hiding one's IP address, as this can potentially be used to identify an account holder (and thus the subject of an action). Anonymizing IP addresses at the network level is easily employable in any technical solution (such as PawS), as it can be implemented completely separate from the actual privacy infrastructure. However, network anonymity does not prevent the identification of a particular user through personal information that is stored or transmitted as part of the application. One alternative is the use of pseudonyms, which allow for application-specific personalization without having to disclose the full identity of a user.

The following sections will briefly describe the concepts of mix networks for network anonymity and the use of pseudonyms in personalization systems, two mechanisms that can be used with our privacy infrastructure. Note that anonymization techniques specific to *location privacy* are discussed in section 5.2 below.

Mix Networks

One of the most popular means of hiding the IP address have been anonymizing proxies, such as `www.anonymizer.com`. The basic idea is to route all communication requests through the proxy, which strips the originating IP address and proceeds to make the connection on behalf of the requesting client. Replies can be associated with the correct request with the help of a lookup table and returned directly and transparently to the client. However, anonymizing proxies have the drawback of providing a single point of attack (or failure) for associating a certain request with a user (or his or her IP address).

A more robust solution is the use of a *mix network*, which routes user requests through a large number of mix nodes before one node finally connects to the desired address, thus making it much more difficult to resolve the user's IP behind a specific request. Figure 3.6 illustrates the concept of a mix network as initially proposed by Chaum [60]. Using public key cryptography, the sender does not send the message directly to the destination address but instead repeatedly encrypts it with the public key of a *mix node*, a publicly known computer that participates in the mix network. Each time the sender encrypts the original message, the address of the corresponding mix-node is used as a new address, in effect chaining a number of decrypt-and-forward operations.

A publicly available implementation of a mix network is for example

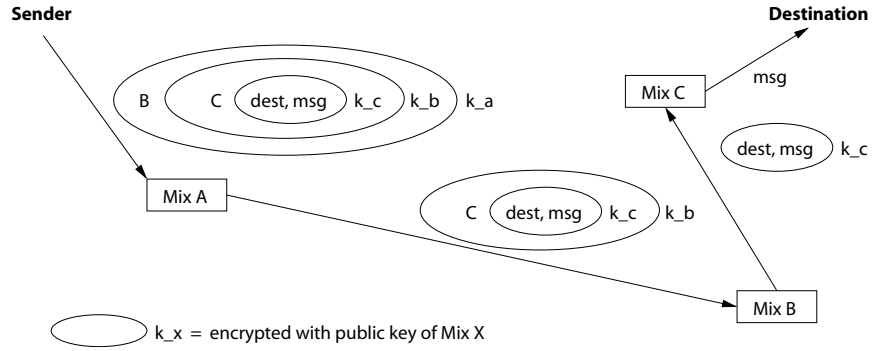


Figure 3.6: *Mix Network Example [60].* The sender repeatedly encrypts the original message with the public key of a random mix, each time prepending the address of the mix. Each mix only knows the subsequent mix to send the packet to, but does not know its contents or any other nodes referenced in the packet.

the "JAP Anon Proxy" at Dresden University.[72] Several projects have improved upon the original concept, most notably the *Crowds* project at AT&T Labs [286], which alleviates the need for preparing a mix-chain (i.e., the repeated encryptions to different mix-nodes) in advance, as it randomly routes messages between mix-nodes.

Pseudonyms in User-Adaptive Systems

Full anonymity does not allow for individual personalization of systems. It therefore becomes often necessary to use *pseudonymous identification*, which allows users to choose a unique but otherwise uncontrolled pseudonym by which he or she can be repeatedly identified in subsequent interactions with the system.

Kobsa and Schreck [196] use mix networks to disassociate user models on arbitrary user modelling servers from actual users, allowing them to provide fully pseudonymous access to both user models and the servers maintaining them.[73] Using traditional role-based access control, clients can be given different access levels to various parts of the user model. Kobsa and Schreck propose a matrix of three by three roles, differing between consumers, producers, and maintainers, as well as between untrusted, trusted, and verified clients.

Since user models not only include buying habits or news preferences,

[72] See `anon.inf.tu-dresden.de`

[73] The location of a user model could in theory provide hints to the actual user, especially when the model is maintained on a private system, e.g., as part of a wearable computer.

but also encompass general user preferences such as privacy, having a solid pseudonymity architecture that allows multiple applications to share a common user privacy model, yet provide robust pseudonymity with respect to user identity, is highly desirable.

3.3.3 Transparency and Trust Tools

Transparency and trust tools are meant to increase consumer trust in a transaction or data exchange, by providing additional background information about the transfer, its conditions, and the parties involved. They link directly into our previously identified social mechanism of trust, as they can provide assurances upon which users can make trust decisions due to incomplete knowledge about their interaction partner.

Transparency tools can range from a single assertive statement, called a "seal" (as it is typically authenticated using some form of digital signatures), to a complex meta-description of a transaction, often called a "social protocol" [83]. After briefly summarizing existing seal programs, we want to focus on a particular transparency protocol called "Platform for Privacy Preferences Project" (P3P), as it will form the basis for much of our privacy architecture in chapter 4.

Trust Seals

Hu et al. [169] classify the goals of trust seals into five different categories: providing privacy assurances; making security assertions; demonstrating consumer satisfaction; expressing reliability; and offering guarantees. Some better-known examples for such seal-classes are the TrustE-seal,[74] VeriSign,[75] BizRate,[76] BBBOnline,[77] and the AOL Merchant Certification program.[78]

While seals have been found to be an effective tool for increasing consumer trust [198], their biggest advantage is also often their biggest drawback, especially when used as the sole source of information: While seals allow complex assurances to be condensed into an easily recognized statement (typically a graphical logo), the actual assurance behind such a seal is often not clear to the consumer. The privacy-assuring TrustE-seal, for example, simply provides assurance that the merchant

[74] See `www.truste.com`
[75] See `www.verisign.com`
[76] See `www.bizrate.com`
[77] See `www.bbb.org`
[78] See `www.aol.com`

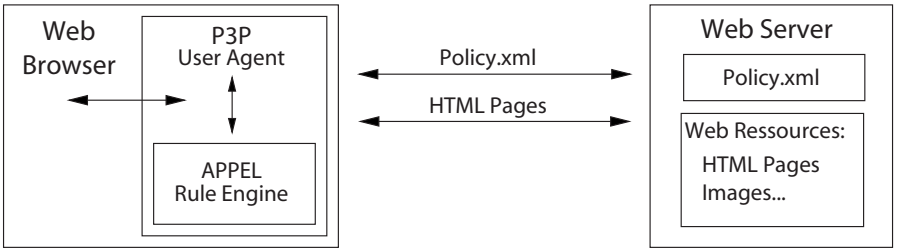


Figure 3.7: *P3P interaction scenario.*

displaying it is honoring its published privacy policy, yet does not qualify in any way the policy itself. Roßnagel calls this "Good notices of bad practices" [299].

The Platform for Privacy Preferences Project (P3P)

The Platform for Privacy Preferences Project (P3P) was launched in May 1997 at the World Wide Web Consortium (W3C) in an effort to develop a specification for automated privacy discussions. It tries to provide the means to communicate – in an electronic form – answers to questions such as "Who will get my personal data?", "Why is this data being collected?", or "How long will my information be stored?". While such information has long been already present in separate natural language statements (as fine print on the back of paper forms, or via a hyperlink off the entry page of a Website), P3P now allows automated processes, e.g., the user's Web browser, to read such information in a machine readable format and provide customized summaries to the user, or even take automated decisions (e.g., whether to block or allow placement of a certain cookie) on behalf of the user [204].

Cranor and Reagle [83] call P3P a social protocol because it mediates interactions between humans, in contrast to technical protocols, which facilitate machine to machine communication. Another example for such a social protocol – and an early inspiration for P3P – is the Platform for Internet Content Selection (PICS) [287]. PICS allows content provider to rate their Internet offering along a specific rating system,[79] embed this information as part of the HTML-code into a Web page, and have Web browsers or search engines automatically parse this information (and act on it). One popular rating vocabulary

[79]The standard does not prescripe a particular rating system, but defines a mechanism to define and reference one. This allows it to be used in almost any kind of content filtering situation, as the actual rating scheme can (and must) be defined by a third party.

is the RSACi system,[80] which rates language profanity, nudity, sex, and violence, each on a scale from 0 to 4. A PICS-compatible browser can use such information to block pages containing content that parents deem unsuitable for their children.

P3P uses a similar approach: Web sites collecting data from their online-visitors and/or online-customers can label their data collections (e.g., a page containing an HTML-form, or even an entire site) with their data collection practices, and consumers can set their browsers to automatically advise them of collection practices that do not conform to their preferences, e.g., using a preference language such as APPEL [80].

The P3P working group[81] is currently improving the original 1.0 specification into P3P 1.1, while keeping backward compatibility with P3P 1.0. This is achieved by introducing all new syntax using the P3P 1.0 extension mechanism [79]. PawS is based on P3P 1.0, but should work with P3P 1.1 as well. The following sections will describe the P3P 1.0 specification in more detail, in order to prepare for the description of our PawS extensions in chapter 4.

P3P Syntax

Figure 3.8 shows an example policy in P3P syntax. P3P uses XML to encode a fixed *vocabulary* that can be used to describe privacy practices, such as the purpose and recipient of a data collection (e.g., lines 24–25) or the duration of data storage (e.g., line 26). A *base data schema* (described in detail in the following section) provides a common set of data elements to reference the individual user data elements these practices apply to (e.g., lines 27–35).

A P3P 1.0 policy (`POLICY`) is enclosed in a `POLICIES` element that allows a single file to hold multiple policies, each being uniquely identified by a `name` attribute (see section 3.3.3 below for details on referencing policies). Each policy is comprised of an `ENTITY` declaration, an `ACCESS` declaration, an optional `DISPUTES-GROUP`, and one or more `STATEMENTS`.

The `ENTITY` block describes the data collector, typically using base data schema elements to give the collector's name and address (lines 4–

[80] RSACi was devised by the Recreational Software Advisory Council in 1996. The council has since been folded into the new Internet Content Rating Association, though the RSACi system is still supported by a number of Web browsers, most notably Microsoft's Internet Explorer.

[81] See `www.w3.org/P3P`

```
<POLICIES xmlns="http://www.w3.org/2002/01/P3Pv1">
  <POLICY name="OnlineShopping"
          discuri="http://www.store.example.com/privacy/privacy.html">

    <ENTITY>
      <DATA-GROUP>
        <DATA ref="#business.name">Example Store</DATA>
        <DATA ref="#business.contact-info.postal.street">Main Street 101</DATA>
        <DATA ref="#business.contact-info.postal.city">Exampletown</DATA>
        <DATA ref="#business.contact-info.postal.postalcode">98103</DATA>
        <DATA ref="#business.contact-info.postal.country">Anotherland</DATA>
        <DATA ref="#business.contact-info.online.email">info@store.example.com</DATA>
      </DATA-GROUP>
    </ENTITY>

    <ACCESS><ident-contact/></ACCESS>

    <DISPUTES-GROUP>
      <DISPUTES resolution-type="independent"
                service="http://www.customerprotection.example.net"
                short-description="CustomerProtection">
        <REMEDIES><correct/></REMEDIES>
      </DISPUTES>
    </DISPUTES-GROUP>

    <STATEMENT>
      <CONSEQUENCE>We use this information when you make a purchase.</CONSEQUENCE>
      <PURPOSE><current/></PURPOSE>
      <RECIPIENT><ours/></RECIPIENT>
      <RETENTION><stated-purpose/></RETENTION>
      <DATA-GROUP>
        <DATA ref="#user.name"/>
        <DATA ref="#user.home-info.postal"/>
        <DATA ref="#user.home-info.telecom.telephone"/>
        <DATA ref="#user.home-info.online.email"/>
        <DATA ref="#user.login.id"/>
        <DATA ref="#user.login.password"/>
        <DATA ref="#dynamic.miscdata"><CATEGORIES><purchase/></CATEGORIES></DATA>
      </DATA-GROUP>
    </STATEMENT>

    <STATEMENT>
      <CONSEQUENCE>We tailor our site based on your past visits.</CONSEQUENCE>
      <PURPOSE><tailoring/><develop/></PURPOSE>
      <RECIPIENT><ours/></RECIPIENT>
      <RETENTION><stated-purpose/></RETENTION>
      <DATA-GROUP>
        <DATA ref="#dynamic.cookies"><CATEGORIES><state/></CATEGORIES></DATA>
        <DATA ref="#dynamic.miscdata"><CATEGORIES><preference/></CATEGORIES></DATA>
      </DATA-GROUP>
    </STATEMENT>

  </POLICY>
</POLICIES>
```

Figure 3.8: *Example of a P3P policy.* An online store collects personal information for site tailoring, as well as during an actual purchase. An independent agency can be contacted in order to resolve disputes.

Access type	Explanation
<nonident/>	The site does not collect identified information
<all/>	Access is given to all identifiable data
<ident-contact/>	Access is given to identified contact information (such as email or postal addresses)
<other-ident/>	Access is given to certain other identified user data, e.g., online account charges
<contact-and-other/>	Access is given to identified contact information (online and postal), as well as other identified information
<none/>	No access to identified data is given

Table 3.2: *P3P access information.* P3P forces data collectors to declare the kind of access they offer to identified user data they collected. Notice the differentiation between *identified* and *identifiable* information: Any information that is not correlated to a specific person (i.e., identified) but could potentially be, is called identifiable.

13). The ACCESS element in line describes what kind of access the data collector provides to the collected personal information – in the example above only access to identifiable contact information (e.g., email or postal address) is given (ident-contact). Table 3.2 lists all possible values for the ACCESS element.

An optional DISPUTES-GROUP can hold one or more DISPUTES elements, which describe dispute resolution procedures that may be followed in case a consumer disputes a service's privacy practices. It lists the type of dispute resolution possible (e.g., contacting customer service, or an independent consumer organization) and a URI of the Web page containing further details on the procedure (e.g., email addresses or telephone numbers to contact).

STATEMENTS contain the individual data elements that the service collects, together with detailed collection practices regarding the purpose, recipient, and retention time of the data collection. An optional CONSEQUENCE element can be used to supply a human-readable description of the effect of the data collection (e.g., to improve customer experience, or to ship ordered items). Tables 3.3, 3.4, and 3.5 list the available values of each of these elements.

The actual data elements that are to be collected are given inside a DATA-GROUP element (see lines 27–35 in figure 3.8 on page 95). The elements that can be specified are defined in the P3P base data schema, which is described in more detail in the following section.

An important part of the P3P syntax is played by the EXTENSION

Purpose type	Explanation
<current/>	The most versatile purpose element: it declares that the data is collected for the user's *current activity*, e.g., to return search results, give access to an online address book, or renew a subscription.
<admin/>	Data is used to administer and maintain the Web site and its computer system (e.g., Web access logs).
<develop/>	The information may be used to enhance or review the site, service, or product, but does not include individual content tailoring to a specific user (see specific purposes below).
<tailoring/>	Information is used for a one-time customization of the site, without retaining this for future use, e.g., when suggesting additional items of interest based on the contents of the user's shopping basket.
<pseudo-analysis/>	The information may be used to build a record of the customer *without attempting to identify* this particular customer, e.g., when trying to understand the interests of different types of visitors.
<pseudo-decision/>	While not trying to identify a particular individual, the collected information might be used to make a decision that directly affects the user, e.g., when modifying the displayed pages based on previous visits.
<individual-analysis/>	Data is collected to research, analyse, and report on the habits, interests, or other characteristics of an individual.
<individual-decision/>	The collected information is used to make a decision that affect the user, e.g., to offer special sales items based on the individual profile of a customer.
<contact/>	The information is used to contact a user for marketing a specific product or service using any other communication channel than the phone (see `telemarketing` below). This does not include replying to specific question of a consumer – in those cases, `current` would be the appropriate declaration.
<historical/>	Information is preserved for historial purposes, as governed by existing laws or policy. Details must be declared in a corresponding `DISPUTES` element, including a link to a human-readable description about the type of historical research planned.
<telemarketing/>	The information is used to contact visitors for marketing a specific product or service via telephone.
<other-purpose>...</>	Any other purpose that is not covered by the above definitions. A human readable explanation must be provided.

Table 3.3: *P3P purpose declaration.* Each data collection statement in P3P must include a `PURPOSE` element, which describes the purpose for which the set of data is being collected [81].

Recipient value	Explanation
<ours/>	Only the data collector ("ourselves") and its agents receive the data, or the data collector is itself an agent, in which case the entity for which it acts as an agent might also receive it.
<same/>	Recipients are legal entities that follow the same data collection practices as the original data collector.
<other-recipient/>	Recipients are legal entities that are accountable to the original data collector, but who may follow different practices.
<delivery/>	Data is shared with entities performing delivery services that have unknown or differing practices.
<public/>	Information is published in public fora, e.g., bulletin boards or directories.
<unrelated/>	Data is shared with unrelated third parties whose practices differ from or are not known to the orginal service provider.

Table 3.4: *P3P recipient information.* Should the data collector share the information with other parties, it must declare its relationship to these parties and/or their status using the RECIPIENT element.

Retention type	Explanation
<no-retention/>	The information is not retained, only used briefly in the course of the stated service.
<stated-purpose/>	Information is retained according to the retention policy, and only as long as necessary for the stated purpose.
<legal-requirement/>	The information is retained for the stated purpose, but might be stored longer due to legal requirements, as stated in the retention policy.
<business-practices/>	Data is retained according to the stated retention policy.
<indefinitely/>	The information is retained for an indeterminate time (e.g., when posting to a public forum).

Table 3.5: *P3P retention information.* P3P offers only a few basic values for describing the retention period – specific times must be given through a human-readable page linked from the corresponding human-readable privacy policy page.

element. It allows P3P to be arbitrarily extended, e.g., for adding application-specific information or future backward-compatible extensions to the standard.[82] Extensions can safely be ignored by user agents not familiar with the particular extension, unless the attribute optional="no" is given, in which case the extension is mandatory, and user agents not understanding it must ignore the whole policy. Figure 3.9 gives an example from the new P3P 1.1, where the new JURISDICTION element, describing the regulatory environment in which the policy is placed, is introduced as an optional extension within the

[82] All new features of the upcoming P3P 1.1 specification [79] have been introduced using this extension mechanism, thus preserving policy backward-compatibility for older P3P 1.0 clients.

```
<RECIPIENT>
  <ours/>
  <EXTENSION optional="yes">
    <JURISDICTION
     service="http://europa.eu.int/smartapi/cgi/sga_doc?smartapi!celexapi!prod!
              CELEXnumdoc&lg=EN&numdoc=31995L0046&model=guichett"
     short-description="31995L0046 Official Journal L 281, 23/11/1995 P. 0031 - 0050">
       Directive 95/46/EC of the European Parliament and of the Council
       of 24 October 1995 on the protection of individuals with regard
       to the processing of personal data and on the free movement of
       such data
    </JURISDICTION>
  </EXTENSION>
</RECIPIENT>
```

Figure 3.9: *Example of a P3P extension.* The P3P 1.1 specification introduces the new `JURISDICTION` element as an optional extension, i.e., P3P 1.0 clients can safely ignore this information and continue to parse the policy.

`RECIPIENT` element.

The extension mechanism will be used in our PawS architecture to extend P3P policies for ubiquitous computing environments (see section 4.2).

P3P Data Schemas

In order to have clients and Web servers talk about the same things when describing privacy policies, P3P needs not only a *vocabulary* defining access policies, purpose declarations, and statements, but also an actual list of data elements that these policies apply to. The list of data elements known to P3P clients and servers is called the *P3P base data schema*.

The base data schema is organized hierarchically, using a dotted notation. This allows for more compact policies, as a number of subelements can be described in a single top element. For example, declaring that a policy applies to the `user.home-info.postal` element is equivalent to declaring all its subelements individually: `user.home-info.postal.name`,[83] `user.home-info.postal.street`, and so forth for the `.city`, `.stateprov`, `.postalcode`, `.country`, and `.organization` subelements. P3P 1.0 defines four top-level elements: `dynamic` (for dynamic data such as clickstream data, cookies, or HTTP headers),

[83] Note that `user.home-info.postal.name` is itself a compound data element, containing a user's given and family name (`user.home-info.postal.name.given` and `user.home-info.postal.name.family`).

Subelement	Substructure	Description	Category
`name`	`personname`	User's name	Physical contact information; Demographic and socioeconomic data
`bdate`	`date`	User's birthdate	Demographic and socioeconomic data
`login`	`login`	User's login information	Unique identifiers
`cert`	`certificate`	User's identity certificate	Unique identifiers
`gender`	unstructured	User's gender (male or female)	Demographic and Socioeconomic Data
`employer`	unstructured	User's employer (company name)	Demographic and Socioeconomic Data
`department`	unstructured	Department or division of employment	Demographic and Socioeconomic Data
`jobtitle`	unstructured	User's jobtitle	Demographic and Socioeconomic Data
`home-info`	`contact`	User's contact information at home	Physical contact information; Online contact information; Demographic and Socioeconomic Data
`business-info`	`contact`	User's contact information at work	Physical contact information; Online contact information; Demographic and Socioeconomic Data

Table 3.6: *P3P user data schema.* Most subelements of the user data schema are in turn divided into more substructures, such as the `contact` data structure. Each element, as well as each structure, features one or more categories, which can be used to simplify the formulation of rules over privacy policies.

`user` (for data such as name, birthdate, or home or work addresses), `thirdparty` and `business` (for name and contact information of third-party and data collector, respectively). The top-element with the most subelements is the `user` data set. Table 3.6 lists the immediate subelements of the `user` data element, with most of these in turn being compound data elements with further subelements. For a complete list of elements, see [81].

In addition, each data element, including abbreviated top elements such as `user.home-info`, have one or more corresponding *categories*. By assigning a category to a data element or a data structure, both services and users can refer to an entire category of data elements when formulating privacy policies or preferences. This also facilitates the

introduction of new data elements, as users might already have preferences regarding a certain type of data elements, even if a particular element is not part of the P3P base data schema. While some of these categories are specified as part of the base data schema definition and cannot be overridden (called *fixed*-category elements), other elements, such as generic date identifiers or dynamic data such as cookies, have so-called *dynamic* categories, i.e., service provider have to declare their categories upon usage. For example, a policy declaring the collection of a cookie containing a user ID would need declared the cookie as belonging to the `uniqueid` category.

P3P Protocol

In addition to machine-readable privacy policies, the P3P specification also defines a protocol for Web browsers and Web servers that allows them to exchange this information efficiently. Using so-called *policy reference files* (see figure 3.10), data collectors can specify data collection practices for a range of different Web sites and/or Web pages in a single file, which may be located at a well-known location (accessible at `/w3c/p3p.xml` off the root of a Web site), linked from within the HTTP header, or referenced from within an HTML or XML document.

3.3.4 Summary

Technical tools and mechanisms form the plumbing of any comprehensive privacy solution. They provide for important characteristics such as secure communications, anonymous connections, and transparent transactions. The technical mechanisms presented in this section only covered tools that we will later employ in our own technical privacy architecture for ubiquitous computing, and thus represent only a subset of the available systems and algorithms. Others might be equally relevant or could even be substituted for those we plan to be using in our system. The goal of this section was not to give an exhaustive list, but instead hint at the possibilities technology offers, and describe those on which we can build upon.

Note that we also did not cover privacy technology that was specifically designed with ubiquitous computing in mind – we will look at alternative solutions in chapter 5, when we compare PawS with other ubiquitous computing privacy frameworks.

```
<META xmlns="http://www.w3.org/2002/01/P3Pv1">
  <POLICY-REFERENCES>
    <EXPIRY max-age="1209600"/>

    <POLICY-REF about="/P3P/policy1.xml">
      <INCLUDE>/*</INCLUDE>
      <EXCLUDE>/cgi-bin/*</EXCLUDE>
      <EXCLUDE>/servlet/*</EXCLUDE>
    </POLICY-REF>

    <POLICY-REF about ="/P3P/policy2.xml">
      <INCLUDE>/cgi-bin/*</INCLUDE>
      <INCLUDE>/servlet/*</INCLUDE>
    </POLICY-REF>

  </POLICY-REFERENCES>
</META>
```

Figure 3.10: *Example of a P3P Policy Reference File,* which allows data collectors to associate certain parts of a Web site with a specific policy. The above declaration associates `policy1.xml` with all contents on the Web site, except for the directories `/cgi-bin/` and `/servlet/`, for which `policy2.xml` holds.

3.4 Guiding Principles

Having reviewed a comprehensive set of mechanisms at our disposal, we will close this chapter with developing our set of guidelines that will govern our own development of a technical privacy-support tool for ubiquitous computing in chapter 4.

As the previous chapter has shown, ubiquitous computing is a powerful vision that has the potential to significantly alter our everyday privacy. If we want to preserve, or even improve upon, today's status quo, we need to explicitly develop guiding principles that can help us develop privacy friendly systems. Otherwise, as section 1.3 above has shown, neither designers nor developers of ubiquitous computing systems will likely make privacy an explicit part of their systems. This section tries to describe such principles, based on our analysis in chapter 2 and the tools reviewed in the previous sections. However, before we set out, we must focus on what exactly we are trying to accomplish, especially given some of the more critical views of privacy set forth in section 2.1.3 above.

In particular, this means that we are *not* trying to achieve total se-

curity, let alone total privacy. Undoubtedly, professional surveillance by spies and private investigators will continue to happen, just as it has happened in the past. New technologies may be found that will be able to (partially) sniff out such surveillance devices. Eventually, better surveillance methods will counter this advantage again. The fact that there have been and always will be a few rotten apples will not spoil the whole batch of technical possibilities ahead for us.

What we can and will be able to achieve is prevent unwanted accidents – data spills of highly personal information that people who have never asked for it suddenly find at their doorstep. What we can do is allow people who *want* to respect our privacy to behave in such a way, so that we will eventually be able to build a long lasting relationship based on mutual trust and respect. And what should also be within our reach is achieving a good balance of convenience and control when interacting with ubiquitous, invisible devices and infrastructures.

Following the Fair Information Practices and their recent enhancements through the enactment of the EU Directive (see section 3.2.1 above), we can identify seven main areas of innovation and system design that future research in ubiquitous computing will need to focus on in order to preserve today's privacy levels for their users. The next sections will elaborate on each of the concepts, ranging from the fundamental notion of notice and consent to the more general non-technical practices such as data minimization and use limitation.

3.4.1 Notice and Disclosure

The most fundamental principle of any data collection system (and ubiquitous systems will, in some respect, play such a role) is the principle of Openness, or simply Notice. In most legal systems today no single data collection – be it a simple id tracking activity or a full fledged audio visual recording – can go unnoticed of the subject that is being monitored (that is, as long as the subject can be personally identified).

Again, ubiquitous devices will per definition be ideally suited for covert operation and illegal surveillance, no matter how much disclosure protocols are being developed. It will always take special detection equipment to be reasonably sure that a certain room or area is not being overheard by others. But openness goes a long way when we want to prevent the mass-market "smart" coffee cup to turn *inadvertently*

into a spy-tool par excellance. Imagine the casual user of a memory-amplifier-coffee-cup accidentally leaving her cup in her colleagues office – only to find in the evening that her colleague has spent most of the day gossiping about her, completely unaware of the spying coffee cup. Even though such accidental recordings for the most part cannot be upheld in courts, the damage is done and the social implications far outweigh the legal ones under such circumstances.

What would be helpful is some kind of announcement system, very much like a radio traffic announcement system, where car stereos will interrupt the playing of a CD or tape if an important traffic announcement comes up. Other analogies would be the `robots.txt` file on World Wide Web servers which allows Web robots to check for the "house rules" before excessively traversing a site, or the well-known emergency frequencies for radio communications that are reserved and constantly monitored for emergency communications. All these examples have in common the notion of a well-known mechanism, a well-known location for the publication of information. Clients interested in this particular information do not need to spend time and energy on searching for it, they can readily access it should such information be available (given that they know about the well-known location for publishing it).

Depending on the type of device, different announcement mechanisms would need to be found. Constant radio broadcasts, for example, would rapidly drain battery of small mobile devices, while it would be perfectly acceptable for rooms and buildings to ceaselessly announce such information. RFID tags could be used to passively announce data collection without using any batteries at all. The restricted storage size of such labels could be enhanced by outsourcing such information to a publicly available Web site and linking to it by merely placing its URI on the label.

The format of such an announcement would be similar to the machine-readable privacy policies of the *Platform for Privacy Preferences* project [81]: Just as P3P allows Web sites to describe their data collection practices in a machine readable way that can then be read and displayed by P3P-enabled browser software (see section 3.3.3), our announcement mechanism would do the same for smart environments, working not with Web browsers but maybe with wearable, wireless user agents that would automatically process queries for personal information according to the user's preferences.

Obviously, power consumption and connectivity problems in the field

of ubiquitous computing will make it difficult to directly reuse results from Internet research projects. However, the main merit of this work lies in the carefully crafted privacy policy vocabulary: using XML as the encoding format, more than a dozen elements allow Web sites to accurately describe the data they collect, the purpose for doing so, the recipients of the data, their retention, and any dispute mechanisms they have in place in order to deal with customer complaints. The difficulties of coming to a consensus for a vocabulary that is acceptable to both privacy advocates and industrial marketers alike probably accounts for much of the three years the P3P project has taken.

Using a declaration format like P3P and announcing it via one or more well-known mechanisms would form the bottom line for any privacy-aware ubiquitous system. Depending on the actual setup of the system, a single announcement might cover a multitude of devices. For example, an office building might make such an announcement for all of the devices that are installed inside, whenever someone enters through its front doors. Rooms in the building might repeatedly reference this main declaration for all sensors or devices the room is equipped with. A wearable system, on the other hand, might be represented by single declaration from its owner's cell phone. Single, autonomous devices that can be operated independently of such central services would require their own announcement capabilities. For example, a future coffee cup with a sophisticated memo function would need to be able to announce its data collection practices even in the absence of any central unit the holder might wear (as long as the cup would actually collect any data without such a central unit).

Not every single device would need to be identified in such an announcement. The goal is to exhaustively enumerate all *types* of data collected, not the individual devices doing so. It does not really matter how many sensors record audio data in a certain room - the fact that audio recording is done at all is the important information. Collation is always possible, and overstating the actual data collection perfectly legal. An office building could collectively declare that audio recording is done in all of its room, even if not all of them actually had sensors equipped. It is up to the owner of the device or system to decide if such overstatement is in her best interest. Of course, certain practices might not be legal in most countries, which place severe restrictions on surveillance such as wiretapping or video recording (see more about that in the use limitation section below).

3.4.2 Choice and Consent

With the enactment of the EU Directive that refined and extended the well-known Fair Information Practices, it is not enough anymore to simply *announce* and *declare* data collection - it also requires collectors to receive *explicit consent* from the data subject. The Directive thus effectively prohibits any collection and usage of personal information, except for certain legal procedures (law enforcement, public health, etc) or when explicitly consented by the individual.

The most common form of explicit consent nowadays is still the written contract. By showing the signature of the data subject under a corresponding piece of text, collectors can in most cases effectively demonstrate that they have received the explicit consent of the subject. In the world of electronic transactions, however, explicit consent is not that easy to come by.

Even though digital signatures based on public-key cryptography are a well established concept, the actual usage of such signatures is still in its infancy. So far, no public-key-infrastructure (PKI) has actually achieved widespread usage, which makes the actual verification of signatures, as well as their revocation, difficult.

But it is not only a question of authenticity that makes digital signatures hard to use, it is also the requirement of explicitness: A certain statement may very well be signed with the secret key of a certain individual, but had the individual actually any knowledge of signing that particular statement, or was it her personal software agent that handled the task in the background, without the user's knowledge?

In electronic commerce, such explicit consent is often achieved by requiring the press of a button to initiate data transfer. In a ubiquitous computing setting, a press of a button might not only be physically impossible (because none of the devices present support a tactile interface), it might also be unusable: With hundreds of devices from a multitude of collectors constantly querying my information as I walk down a busy street, pressing the OK button on my cell phone every time I want to authorize transfer will surely annoy even the most patient person.

Another often overlooked problem the notion of consent poses to system design is the requirement of choices: With only one option available, getting consent comes dangerously close to blackmailing. Imagine that in order to enter a public building, you must agree to completely

unacceptable practices. Certainly you could always walk away from such a deal, but can you really?[84]

In order to make consent a viable option, more than the "take it or leave it" dualism must be offered. Office buildings could offer me to track my position within the building in order to offer customized navigational services. If I choose to decline, it must be possible to selectively disable the tracking functionality without either shutting down the whole system for all other visitors, or me not entering the building.

Advancements in audio and video processing might make such choices available for selective recordings: Instead of requiring all participants of a meeting to consent to a comprehensive audio or video recording, the system could only track those who agree to the recording, while the voices of all others will be muted, their picture on videos anonymized. A simple solution along similar lines was used in the Classroom 2000 project at Georgia Tech, where classroom recordings would focus on the teacher and his replies, while voices and faces of students where deliberately of low quality [3].

3.4.3 Anonymity and Pseudonymity

Given the difficulties in asserting explicit consent in electronic communications, one viable alternative to personal data collection are the notions of anonymity and pseudonymity. Not only are they an important option when offering clients a number of choices (so that those who wish to remain anonymous can remain so), they also allow the legal collection of certain types of data without requiring user consent.

Anonymity can be defined as "the state of being not identifiable within a set of subjects." The larger the set of subjects is, the stronger is the anonymity [263]. A large number of both free and commercial anonymity services are already in widespread use on the World Wide Web. Using anonymizing proxies, for example the popular Web service `www.anonymizer.com`, or more sophisticated "mixes", like the "Freedom" software product of the Canadian software company Zero-Knowledge, Internet users can already today hide their IP address from the Web site hosting the accessed page.

[84] Some might argue that this is no different from most supermarkets today, which already feature a comprehensive video surveillance system. In most legal systems, such surveillance is possible under very restrictive guidelines that place restrictions on purpose, use, and retention of such video feeds.

Even though the technology behind such services is already well established, such methods might not be feasible in a ubiquitous computing environment. Communications between small ubiquitous devices will often happen in a much more dynamic environment, where long chains of communication (like they are used in mixes) might not last long enough because devices constantly enter or leave the scene. Direct communications on the other hand often disclose my real identity, unless wireless protocols would be adapted to use one-time addresses instead of their fixed hardware (MAC) address (as it is done in the Bluetooth standard). Sensing hardware is also different from network cards: My real-world appearance, unlike my cyberspace one, cannot be disguised that easily – any video camera can get a clear enough shot of me if it's pointed at my face.

Anonymity has also disadvantages from an application point of view. Being anonymous prevents the use of any application that requires authentication or offers some form of personalization. Pseudonymity is an alternative that allows for a more fine grained control of anonymity in such circumstances: by assigning a certain ID to a certain individual, this person can be repeatedly identified until she changes to a different ID. Using the same pseudonym more than once allows the holder to personalize a service or establish a reputation, while always offering her the possibility to step out of that role whenever she wishes.

Whether anonymous or pseudonymous – if data cannot be traced back to an individual (i.e., if it is unlinkable), the collection and usage of such data poses no threat to the individuals privacy. Consequently, legal frameworks such as the EU Directive lay no restriction on the collection of anonymous (or pseudonymous) data. Determining when certain type of information can be linked back to a person, however, is more often than not subject of debate. For example, even randomly generated pseudonyms might be linkable under certain circumstances: In case a pseudonym is used in conjunction with a certain fact that is easy to identify in a sufficiently small set, linking becomes trivial. An active badge might be programmed to change its ID every five minutes, though the fact that the tracking system is able to exactly pinpoint its location would make this change obvious (and thus linkable) in the logs. Alternatively, combining pseudonymized location information with background information about a particular individual's office address or favourite restaurant can easily result in a complete de-anonymization of the data [36].

Data-Mining technology allows much more remote coincidences to be assembled into a single coherent picture, therefore greatly increasing the potential of *any* type of information to be used for linking. Although German privacy-commissioners have argued for placing severe restrictions on the use of data-mining applications [135], their call might not be realistic.

3.4.4 Proximity and Locality

It seems that our above observations regarding the feasibility of certain desirable aspects in a privacy-aware ubiquitous system – such as clear notices, explicit consent, and unlinkable pseudonymity – might prove too difficult for efficient and reliable implementation. One possibility to face this technological reality while still preserving some desirable state of protection, even when this means some form of sociological adjustment, are the principles of proximity and locality.

The idea of proximity is basically a practical solution to much of what makes notice and consent hard. Instead of announcing each and every data collection, taking care to get the required consent, and handle those frequent cases where various people do not give their consent, imagine the following: Future societies (and with it the legal system) will accept the fact that personal gadgetry (like coffee mugs or "smart" clothing) can record conversations and behaviors *whenever its owner is present*. Just as if people would never forget a thing they witnessed. Note that this does not mean that people would suddenly be omniscient – their memory prosthesis (i.e., their coffee mugs) would only grant them the gift of indefinite recollection (currently most legal systems treat any recording without the explicit consent of all parties as surveillance, which is only allowed by law enforcement in certain, court-ordered situations). In case the owner would accidentally leave such a device so that it could witness a conversation or meeting of other people in her absence, all sensory equipment would be turned off until the owner's presence would be detected again.

Such a detection mechanism could be simple. Of course, future advanced sensors could use biometry to check if the cup's owner is actually holding it. It could also use the presence of certain IDs in the clothing of the owner as a trigger: Only if a certain predefined signal would be emitted from the owner's wearable computer, its sensors would be operational. The problem would be further simplified if the cup's data

storage would be outsourced to the holder's wearable computer: In this case it would be sufficient to simply check for the presence of any type of outsourcing facility, in effect acting as a collection device for anybody holding the cup (or sitting next to it).

Although this would alleviate a number of technical problems, recording each and every conversation and behavior would be more than just chatting with friends who suddenly have very good memory. Storage also allows your friends playing this information to people unknown to you, who then effectively witness events they were no part of. While one might still be comfortable with the idea of friends having a good recollection of past discussions together, one would certainly be less comfortable with their friends playing their recordings to a group of strangers for entertainment value, giving them not just a summary, but an accurate word for word reproduction.

Along similar lines as the idea of proximity aims the notion of *locality*. Instead of working out complicated authentication protocols that govern the distribution of collected information, so that it is in compliance with whatever recipient information has been previously announced, information could simply be tied to places at which it is collected. Should a table in a room on a ground floor be allowed to ask the flowerpot on the hallway outside to contact the light fixtures in the staircase for the information that the soda machine on the 3rd floor is currently acquiring? Should my printer tell everybody walking by what it is printing at the moment, only to have them pass this information on to the people they meet on the subway or at the airport, until this data ends up on the other side of the world?

In essence, one would require that information is not disseminated indefinitely, even not across a larger geographic boundary, such as buildings or rooms. Information collected in a building would stay within the building's network. Anybody interested in this information would need to be actually physically present in order to query it. Once present, however, no additional authentication would be required anymore – the printer in the hallway would be happy to tell anybody passing by and stopping for a chat which documents (and by whom) were printed on it last night.

This concept resembles privacy protection (or the lack of it) in small, rural communities: Everybody knows everything about each other, and is only too happy to tell. Once someone leaves the boundaries of the village, however, access to information about its inhabitants becomes

difficult, if not impossible. Though word of mouth allows information to travel far beyond the originating locality, the information value drastically decreases with increasing distance.

In such a scenario, observing anything from a larger distance becomes impractical. Even though it is not impossible to acquire certain information, it ultimately requires physical locality to its source. This wouldn't be too far from our current status quo where law enforcement or private investigators routinely interview witnesses for their version of the events – only that coffee mugs and tables cannot talk. Not yet.

3.4.5 Adequate Security

Not surprisingly, talking about privacy almost always leads to security considerations. In most discussions, the significance of the latter is often perceived much higher than that of the former. The idea is tempting: once we solve security, that is, once we are able to achieve authenticity and trusted communications, privacy will be a by-product that follows inevitably from a secure environment.

Secure communications and storage methods have been around for quite some time, and security experts are constantly refining the algorithms to keep up with the rapid technological development. However, ubiquitous devices will introduce a whole new set of constraints, mainly in the areas of power consumption and communication protocols: there is only so much energy to power an embedded processor in, say, a felt pen, that it will perhaps not be enough to compute the product of two 2048-bit prime numbers. And a pair of smart shoes will probably pass a store front in a few seconds, barely enough time to go through with an orderly security protocol for establishing a secure communication.

Even with GHz Desktop power, security experts question if absolute security can ever be achieved. True, 2048-bit public key encryption is probably secure for the foreseeable future. But in order to prevent misuse, keys need to be encrypted by pass-phrases, which invites the usual problem of choosing nicknames of family members or friends, or writing them down next to the keyboard. Smartcards are often hailed as the ultimate personal security device, but these, too, need to be protected from unauthorized use once they fall into the wrong hands. And even if biometrics will ever allow us to use our fingerprints or retinas to replace personal passwords, key distribution and management for tens and hundreds of small and miniature personal devices (everything from

socks to umbrellas to door knobs) will almost certainly challenge the most clever user interface.

We can reduce much of this complexity by employing robust security only in situations with highly sensitive data transfer, such as financial transactions, or the transfer of medical information. In most other cases, the principle of proportionality applies: cracking a 512-bit key might be feasible given the proper hardware, but if cracking the code would mean a reward of only $10, this would hardly be worth the effort. Similarly, sending temperature data from a sensor to its base station might not need to be encrypted at all. After all - if an eavesdropper is close enough to overhear its low-power radio communication taking place, he might as well sense the current temperature by himself.

Here the principle of locality becomes relevant again: if we start broadcasting otherwise innocuous information like temperature or noise levels from a certain local context across many hops to physically distant (or separated) places, we effectively create surveillance devices. If, however, such data is sent only locally and not transmitted further, the lack of encryption is of no concern, therefore simplifying implementations at a reasonable level of compromise.

The important aspect to realize is that security might not be the panacea it appears to be, and it might not need to be that panacea either. If we consequently apply principles like proximity, locality, and proportionality, much of our basic infrastructure could indeed function without any explicit security model at all, while still adequately respecting many of the privacy needs of its users.

3.4.6 Access and Recourse

Trusting a system, and especially a system as far reaching as a ubiquitous one, requires a set of regulations that separate acceptable from unacceptable behavior, together with a reasonable mechanism for detecting violations and enforcing the penalties set forth in the rules. Both topics belong more into the realm of legal practice, where laws and codes of conduct will need to be revised or newly established in order to address the special requirements of typical ubiquitous computing environments.

However, technology can help implementing specific legal requirements such as use limitation, access, or repudiation. Augmenting a P3P-like protocol with something like digital signatures would allow for

non-repudiation mechanisms, where parties could actually prove that a certain communication took place in case of a dispute. Database technology could provide data collectors with privacy-aware storage technology that would keep data and its associated usage practices as a single unit, simplifying the process of using the collected data in full compliance with the declared privacy practices. Sophisticated XML linking technology could enable the data subject direct access to his or her recorded information in order to enable the required access rights.

The principles of *Collection Limitation* and *Use Limitation* set forth in the Fair Information Practices can further simplify such access requirements. As we have seen in section 3.2.2, they require data collectors to

- only collect data for a well-defined purpose (no "in-advance" storage)
- only collect data relevant for the purpose (not more)
- only keep data as long as it is necessary for the purpose

Together with anonymization or pseudonymization, these principles might save both time and effort that would otherwise be spent in order to properly collect, protect, and manage large amounts of sensitive personal information.

3.5 Summary

This chapter provided us with the basic tools to build our own privacy-awareness solution for ubiquitous computing. It gathered primarily support from existing technical solutions (section 3.3) such as encryption and authentication tools (e.g., SSL and digital signatures), transparency and trust tools (e.g., P3P and seal programs), and anonymization and pseudonymization tools (e.g., mix networks). These technologies are already in often widespread use on the global Internet, and we can for the most part readily employ them for in our ubiquitous computing infrastructures.

The legal frameworks and guidelines presented in section 3.2 form our environment, the set of norms that govern the way society has decided to live together. As we have seen, two different approaches exist, and we will explicitly choose to base our technical architecture on the presence of a strong legal support that allows us to implement

part of our solution in "legal code", not technical one. We have also learned that social tools such as moral values, ethical theories, and concepts such as trust, play an important role in any privacy solution.

These tools, together with the changes that ubiquitous computing will bring for our personal privacy (as seen in chapter 2), prompted us to lay down our guiding principles in section 3.4, based on the ideas of the Fair Information Principles and discussed in light of the technical possibilities and interaction modes of ubiquitous computing systems: notice and disclosure, choice and consent, anonymity and pseudonymity, proximity and locality, adequate security, and access and recourse. These principles will be our yardstick by which we will evaluate our proposed infrastructure in chapter 4 below.

4 PawS – A Privacy Awareness System

Cooperation, like other difficult things,
can be learned only by practice.
John Stuart Mill[1]

In the previous chapter we showed that a large number of tools exist that we can use to build privacy-respecting ubiquitous computing systems, both in terms of technology and through societal means, such as laws and moral codes of conduct. We also discussed in section 3.2.4 how we envision the division of labor between these components: Not a perfect protection of personal data through rigorous employment of digital rights management systems, but instead an ability to easily have our ubiquitous computing systems "do the right thing" right from the start. Transparency and accountability tools are not designed to prevent the abuse of personal data through malicious parties, but can help respectable collectors of our personal data to use our information in accordance with our preference.

This chapter now presents in detail the architecture and implementation of PawS, a technical tool designed to complement existing (and future) legal codes, social rules, and moral norms in order to provide privacy in future ubiquitous computing environments. PawS does not aim at being a complete tool, nor being particularly perfect in its usage. Its user interface and feature set would undoubtedly benefit from a rigorous user study, its code base could be improved through more thorough testing. But this is not the main focus of PawS. It is thought of as a proof of concept, a hint at how any such future system might look like, a thought experiment on how technology can supplement a comprehensive legal protection, and last not least as a tool for refining the boundaries between technology, law, and social norms.

We will set out by summarizing the requirements for such a system developed in the previous chapters, notably sections 1.1, 2.2 and 3.4,

[1] in "Civilization: Signs of the Times," Dissertations and Discussions, vol. 1, 1836

before briefly enumerating the related technologies and projects that PawS builds and improves upon. Sections 4.3 through 4.5 describe the PawS architecture in detail, specifically its three main components: *Privacy Proxies*, *Privacy Beacons*, and the *Privacy Database*. As usual, we will end this chapter with a discussion of the presented topics, specifically we will try to judge the merits of such a system given our previous analysis in chapter 2.

4.1 General Overview and Requirements

Figure 4.1 shows an example of PawS in operation: Upon entering a ubiquitous computing environment with a number of available services (here: a print service and a location tracking service using a video camera), a *privacy beacon* (1) announces the data collections of each service and their policies using a wireless communications channel such as Bluetooth or IrDA. In order to save energy, the mobile *privacy assistant* (2) the user is carrying delegates this information to the user's personal

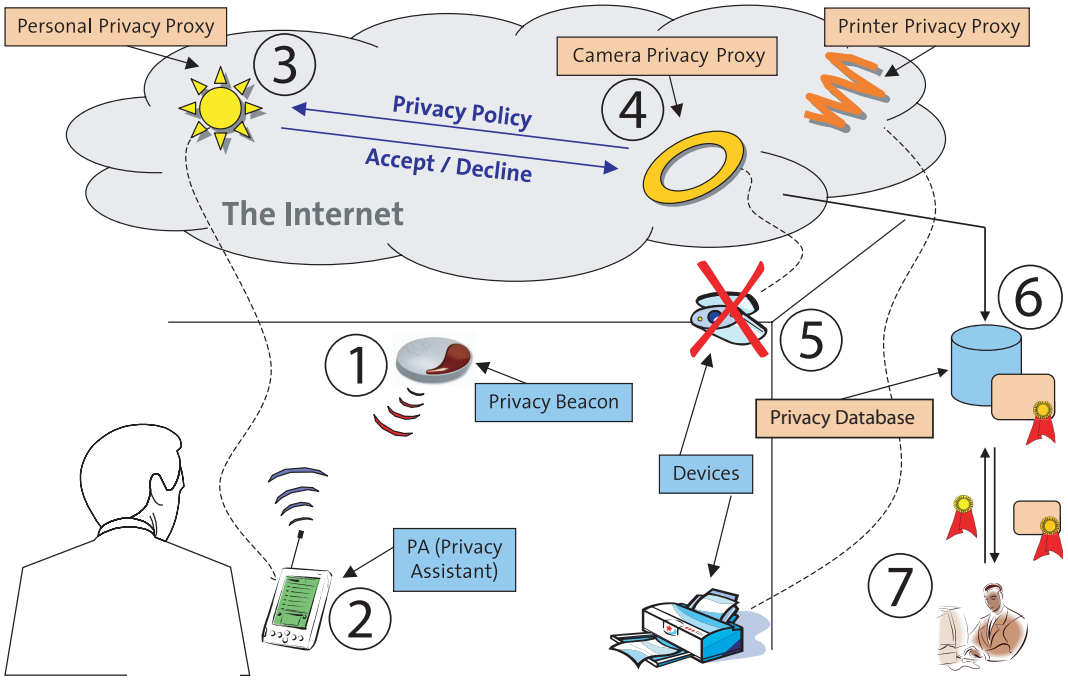


Figure 4.1: *Overview of the Privacy Awareness System:* Upon entering a ubiquitous computing environment with a number of data collections taking place (3,4), optional services can be configured to suit the user's privacy preferences (5). Mandatory data collections (e.g., security cameras) can at least be detected (1) and collection details be recorded (2), allowing users or consumer interest groups to hold data collectors accountable for their statements. Data is stored along with the collection policy (6), allowing later enforcement of purpose and recipient restrictions (7).

privacy proxy residing somewhere on the Internet (3), which contacts the corresponding service proxies at their advertised addresses (4) and inquires their *privacy policies*. After comparing those privacy policies to the user's privacy preferences, the user proxy decides to decline usage of the tracking service, which results in disabling the location tracking service of the video camera (5). Should the individual decide to use the service, the collected data is stored in a *privacy database* along with the original data collection policy (6), thus requiring submission of a matching query policy for every query (7).

In designing the general architecture of such a privacy awareness system, we follow the six principles we set out earlier (section 3.4) for preserving privacy in ubiquitous computing: notice, choice and consent, proximity and locality, anonymity and pseudonymity, security, and access and recourse. As pointed out in the introduction, anonymity, pseudonymity, and security (i.e., secure communication and access) are useful tools when being a supportive part of the infrastructure, but should not be taken as isolated solutions. Consequently, our system employs anonymous and secure connections, as well as reasonable access controls, whenever possible to prevent unwanted data spills and trivial data sniffing. While the system could support the principles of locality and proximity, the focus of the initial prototype lies primarily on implementing the other three principles for use in a ubiquitous computing environment:

- *Notice:* Given a ubiquitous computing environment where it is often difficult for data subjects to realize that data collection is actually taking place, we will not only need mechanisms to declare collection practices (i.e., *privacy policies*), but also efficient ways to communicate these to the user (i.e., *policy announcement*).

- *Choice and consent:* In order to give users a true choice, we need to provide a selection mechanism (i.e., *privacy agreements*) so that users can indicate which services they prefer.

- *Access and recourse:* Our system needs to provide a way for users to access their personal information in a simple way through standardized interfaces (i.e., *data access*). Users should be informed about the usage of their data once it is stored, similar to call-lists that are often part of monthly phone bills (i.e., *usage logs*).

The following sections describe the four core concepts of our system,

which provide us with the necessary functionality to implement the high-level requirements listed above: Machine-readable privacy policies to provide *choice and consent*, policy announcement mechanisms to give *notice*, privacy proxies for supporting *access*, and privacy-aware databases for *recourse*. While *proximity and locality* are not yet addressed in the current prototype, extension mechanisms allow for their implementation once suitable representation techniques have been developed.

4.1.1 Machine-Readable Privacy Policies

As we have seen in the previous chapter, privacy policies are an established principle in legal domains to codify data collection and usage practices. We have also seen in section 3.3.3 how the "Platform for Privacy Preferences Project (P3P)" allows the encoding of such privacy policies into machine-readable XML, allowing automated processes to

```
1  <POLICY name="FollowMe" discuri="http://www.example.org/services/follow-me/">
2    <ENTITY>
3      ...
4      <EXTENSION>
5        <SERVICE name="Follow-me Phone Service" type="continuous" mode="optional">
6          <communication/>
7          ...
8        </SERVICE>
9      </EXTENSION>
10   </ENTITY>
11   <DISPUTES-GROUP> ... </DISPUTES-GROUP>
12   <ACCESS><all/>
13     <EXTENSION optional="yes"> <ACCESS-METHODS>
14       <UPDATE rpc_uri="http://www.example.org/soap/" service_urn="access">
15         <DATA ref="#user.login.password"/> </UPDATE> </ACCESS-METHODS>
16     </EXTENSION>
17   </ACCESS>
18   <STATEMENT>
19     <CONSEQUENCE>Your telephone calls will be routed to you.</CONSEQUENCE>
20     <PURPOSE><current/></PURPOSE>
21     <RECIPIENT><ours/></RECIPIENT>
22     <RETENTION><stated-purpose/></RETENTION>
23     <DATA-GROUP> <DATA ref="#user.login.id"/>
24                  <DATA ref="#user.login.password"/>
25                  <DATA ref="#user.location.current.symbolic.room">
26     </DATA-GROUP>
27   </STATEMENT>
28 </POLICY>
```

Figure 4.2: *Example of a PawS privacy contract* for a follow-me telephone service (abbreviated): Apart from the user's ID and password that has to be submitted when trying to use the service (lines 23-24), the service also (implicitly) collects the user's current location (e.g., room number) through a tracking system (line 25). PawS privacy contracts are P3P 1.0 privacy policies with additional access and service information: Linex 5-8 describe how a user privacy proxy can access the collected data. See section 3.3.3 for details on P3P policies.

read such policies and take actions on them [81]. Figure 4.2 shows an (abbreviated) example of a PawS privacy contract, which is an extended version of a regular P3P 1.0 privacy policy (see section 4.2 below). It contains the XML elements to describe for example who is collecting information (line 2, abbreviated), what data is being collected (lines 15-18), for whom (line 13), and why (line 12). Using a similarly machine-readable preference language such as APPEL [80], users can then express personal preferences over all aspects of such policies and have automated processes judge the acceptability of any such policy, or prompt for a decision instead.

Even though P3P has been developed with the Web in mind, its syntax allows for domain-specific extensions (using the `EXTENSION` tag, see line 5 in figure 4.2) that enable us to use these mechanisms also within the context of a ubiquitous computing environment (*choice and consent*). Specifically, our privacy contracts extend P3P in two areas: the underlying dataschema and the policy itself. The original P3P dataschema needs to be extended to account for sensory data collections (such as cameras, microphones, or floor pressure sensors) and location data, while the policy mechanism itself needs to take our automated access facilities into account. Section 4.2 below will describe our extensions to the P3P base dataschema in greater detail and will explain how we extend P3P policies into *Privacy Contracts*.

4.1.2 Policy Announcement Mechanisms

While P3P is a Web technology and thus uses HTTP-headers as well as well-known URI-locations on each Web server to help user clients locate such policies, we need an alternative mechanism in a ubiquitous computing environment. We can differentiate between two types of data collection that will need different ways of communicating such privacy policies to the data subject (*notice*):

- *Implicit announcement:* In many cases, the user client is actively locating and using a service offered by the environment. In this case, we embed the P3P policy (or links to it) into the service discovery protocol, such as the one in Jini [342] or into the reader-to-tag protocol of an RFID reader.

- *Active policy announcement:* Some services such as audio or video tracking might work continuously in the background, without the

need for user interaction in order to gather data. In this case, a *privacy beacon* constantly announces the privacy policies of implicitly running data collections, using a short-range wireless link.

An example of the first case – embedding privacy policy information into the service protocol – can be found in chapter 6, where we apply our privacy principles in the domain of RFID technology. In our prototype system we have concentrated on the second case: building an explicit policy announcement mechanism through the use of dedicated beacons. These are described in greater detail in section 4.4 below.

4.1.3 Delegating Data Transfer

In contrast to the typical Web data transfer involving form filling and button clicks, data transfers in ubiquitous computing environments will often happen automatically, e.g., upon entering a particular area or performing a certain 'real-world' action. In order to facilitate controlled data transfers under such circumstances, we need concepts and mechanims to authorize certain data transfers beforehand so that the actual transfer can happen without the explicit notice of the user. This is the role of *privacy proxies* in our system.

Privacy proxies handle privacy-relevant interactions between data subjects and data collectors (i.e., policy access and data collection) but also provide access to specific user control capabilities disclosed in the privacy policy such as data updates and deletes, or querying usage logs. Privacy proxies are continuously running services that can be contacted and queried by data subjects anytime, allowing them instant access to their data (see items 3 and 4 in figure 4.1).

Each ubiquitous computing environment either features a single such *service* proxy to handle all its data collections, or multiple service proxies for each individual service it offers. Similarly, each user is expected to have a corresponding *personal* privacy proxy, which handles all interaction between service proxies in order to exchange user data or query their usage logs (in case of disconnects, a mobile device could temporarily act as a substitute for a personal privacy proxy residing on the network).

Privacy proxies are configured using a preference language such as APPEL, described above, typically involving a small set of general rules (which could be created by a trusted third party and downloaded by the user) and a larger set of incrementally created user-specific rules.

As part of such an interaction between user and service proxies, an agreement is made in form of an XML-document containing the data elements exchanged and the privacy policy applying to them (both is encoded in the P3P policy). Such an agreement document also contains an explicit *agreement-id* for later reference, as well as detailed information on how the user proxy can access the service proxy (see our extensions to the `ACCESS` element in figure 4.2, lines 12–17). For example, should the user decide to update her email address with all places that have it on file, her privacy proxy contacts each service's update function to transparently update the changed data (*access*). Section 4.2 describes these agreements, called *privacy contracts*, in more detail.

4.1.4 Policy-Based Data Access

Once data has been solicited from the user (either actively by receiving a data submission via the privacy proxy, or implicitly by receiving sensor data such as video or audio feed), it is stored in a back-end database (see items 6 and 7 in figure 4.1 above). In order to prevent accidental use of information that is in disagreement with the previously granted privacy policy, the database not only stores the data collected, but also each individual privacy policy that it was collected under (i.e., the ist corresponding privacy contract).

By combining both data elements and their respective policy into a single unit managed by the database, we can have the database take care of observing that the promises made in a privacy policy with respect to the lifetime, usage, and recipient of a certain piece of information are kept, as well as provide users with a detailed "usage log" of their personal data (*recourse*). Note that since policies are often invariant for a large number of collected data elements, storing an additional pointer to such a policy only adds a small overhead for storage requirements. Section 4.5 describes our prototypical database with support for privacy contract based data storage and access.

4.1.5 Summary

Our six privacy principles require notice, choice and consent, proximity and locality, anonymity and pseudonymity, security, and access and recourse. Using our toolbox of encryption and anonymization technology described in chapter 3 above, we can provide a base level of security and anonymity using standard Internet technology such as

SSH or mix networks. The focus of our work is on providing notice, choice and consent, and access and recourse, using the following components: machine-readable privacy policies, automated policy announcement mechanisms, delegated data transfer, and policy-based data access.

These four requirements will be supported with the help of four core concepts in our PawS architecture, specifically,

- *Privacy Contracts* that provide a virtual link between collected user data and the privacy policy under which it was collected.
- *Privacy Proxies* that allow the user to delegate his or her privacy preferences to an automated process, which can unobstrusively monitor any attempted or actual data collection and configure it to best suite the user's individual preferences.
- *Privacy Beacons* that enable the user to detect unnoticable data collections through their policy announcement mechanism, and consequently let his or her privacy proxy handle the required data transfer.
- *Privacy Databases* that enable users to access their stored data easily while allowing data collectors to easily follow their own privacy policies.

The following sections will present each concept in greater detail.

4.2 Privacy Contracts

Privacy contracts form the basis for our privacy-aware infrastructure.[2] They are the core element in any kind of data exchange, identifying the type of data collected, the identity of the data collector, and the means to access this information. Most of this functionality is already part of the machine-readable privacy policies of P3P. However, PawS extends the existing P3P policy in order to form privacy contracts that support even more automated data processing and management than P3P alone. This also supports backward compatibility to existing P3P tools, as the policy format remains unchanged. Our privacy contracts are thus P3P 1.0 policy files with a number of ubiquitous computing specific extensions, as it was illustrated in figure 4.2 on page 118.

[2]Privacy contracts were developed as part of the diploma thesis of Mark Stäheli [325].

In a first step, we have added a number of ubiquitous computing specific extensions to the regular P3P base dataschema in order to better support the envisioned data collections in ubiquitous computing environments. These extensions, regarding new perception mechanisms such as cameras and microphones, as well as location information, are described in section 4.2.1 below. Section 4.2.2 then describes the information present in our contracts, before sections 4.2.3 and 4.2.4 detail our two extensions to the P3P policy format, the `ACCESS` and the `SERVICE` elements.

4.2.1 Extending the P3P Base-Dataschema

The P3P Dataschema defines the types of data that can be referenced from within a P3P policy, e.g., when soliciting the user's address data, but also for referencing the data collectors own address in the policies header. In ubiquitous computing environments where personal information could be collected anytime, anywhere, additional data elements need to be defined. Cameras, microphones, and other sensors can record personal audio-video information as well as the user's location. Being primarily a Web standard, the P3P dataschema has no means to express the collection of such elements.

Perception Data

In order to express the first kind of data collections, those done by cameras, microphones, and other sensors (e.g., floor pressure sensors), we add *perception data* to the P3P dataschema. It allows data collectors to specify four types of sensory perceptions: still photos, videos, audio recordings, and miscellaneous sensor data. Table 4.1 shows the four additional data structures (`photo`, `video`, `audio`, and `misc`) and brief examples of their usage.

Each of these data elements is part of the `perception` category,[3] a new P3P category that we can incorporate via the `other-category` extension mechanism (see table 4.2). In addition, any collected data that is marked as `perception` data indicates that not only sensors are involved during the collection, but also that secondary information can potentially be derived from this information. For example, video or audio recordings can easily reveal the age and race of a person (so we

[3]See section 3.3.3 for a description of P3P categories.

New Data Schema Element	Example
`user.perception.submitted.photo`	Passport photograph
`user.perception.submitted.video`	Pre-recorded video
`user.perception.submitted.audio`	Training data for voice recognition
`user.perception.submitted.misc`	Pre-recorded walking patterns
`user.perception.submitted`	Any user-submitted perception data
`user.perception.current.photo`	Live picture snapshot
`user.perception.current.video`	Live video (and audio) recording
`user.perception.current.audio`	Live sound recording
`user.perception.current.misc`	Infrared sensor measuring current body heat
`user.perception.current`	Any preception data that is recorded live

Table 4.1: *Extending the P3P base data schema with perception data.* P3P does not have predefined data elements for describing camera or microphone recordings, let alone sensory information such as a floor pressure sensor. The `perceptionData` element adds data elements capable of describing the collection of still photos, videos, audio recordings, and miscellaneous sensory information.

must include the `demographic` category), and potentially identify this person (which implies the `uniqueid` category).

Even though other physical attributes, such as fatigue or nervousness, could potentially be also derived from such data, we have not included the `health` category, as extracting such information is still too difficult and unreliable to be done routinely. This does not preclude an explicit declaration by the data collector in case such derivative information concerning a subjects physical or mental state is actually used. Audio information is additionally assigned the `content` category, as recorded discussions or speeches are similar to contributions in chat rooms or other on-line forums.[4]

These data elements are in turn subdivided into two classes, `current` and `submitted`. This allows data collectors to indicate in which way the information is solicited from the data subject. While `current` perception data is collected 'on-the-spot', i.e., by sensors such as cameras or microphones in place, some applications might also support user-submitted information, e.g., a voice print or a photograph. Note that in-place collection of sensory information additionally results in location information being revealed, thus prompting the inclusion of the `location` category as part of the `current` data set.

[4]If a video also contains an audio track, the data type `audio` must be declared in addition to `video`.

Categories	photo	video	audio	misc
submitted	perception demographic uniqueid	perception demographic uniqueid	perception demographic uniqueid content	perception
current	perception location demographic uniqueid	perception location demographic uniqueid	perception location demographic uniqueid content	perception location

Table 4.2: *Categories of perception data.* All perception data elements are part of our newly defined `perception` category, as well as part of the `demographic` and `uniqueid` category (as such information can be deduced from audio and video recordings). In addition, audio data is in the `content` category, while all `current` sensory data includes information about the user's current `location` as well.

Location Data

A similar extension is made for location data. Being a Web technology, P3P does not have an explicit representation of location data in its dataschema. PawS defines a `geographicLocData` schema element, which describes a location using typical geographic coordinates such as longitude, latitude (both using `hour`, `minute` and `second` subelements), and altitude (see table 4.3 below). This information could for example be solicited from the user's GPS, or be implicitly determined by a service when interacting with the user at a certain location. An alternative form of location data is the symbolic description of a place, e.g., the city name, street address, or building identifier. This second form of location information simply reuses the existing street-level location data already present in the P3P dataschema, the `postal`-structure (which describes a complete postal address).

In order to regulate the granularity of the location data, users can optionally be asked to indicate the data resolution (in meters, or fractions of a meter) using the `accuracy` data element.[5] Service providers can indicate the level of granularity needed by requesting only selected elements from the complete location set, e.g., only asking for `*.country` if only the user's current country of location is required, or only `hour` and `minute` from a geographical location.

[5]Note the distinction between the terms *accuracy* and *precision*. Precision indicates the exactness of a measurement, i.e., the number of decimal places, while accuracy indicates how close a measured value is to the "real" value.

New Data Schema Element	Description

`user.location.submitted.geographic.longitude.*`
`user.location.submitted.geographic.latitude.*`
`user.location.submitted.geographic.altitude`
`user.location.submitted.geographic.precision`
	Geographical location of the user (self-submitted, e.g., from GPS)

`user.location.submitted.symbolic.name`
`user.location.submitted.symbolic.street`
`user.location.submitted.symbolic.stateprov`
`user.location.submitted.symbolic.postalcode`
`user.location.submitted.symbolic.organization`
`user.location.submitted.symbolic.country`
	Symbolic address of the user (self-submitted)

`user.location.current.wlan.geographic.longitude.*`
`user.location.current.wlan.geographic.latitude.*`
`user.location.current.wlan.geographic.altitude`
`user.location.current.wlan.geographic.precision`
	Geographical location of the user (using WLAN connectivity data)

`user.location.current.wlan.symbolic.name`
`user.location.current.wlan.symbolic.street`
`user.location.current.wlan.symbolic.stateprov`
`user.location.current.wlan.symbolic.postalcode`
`user.location.current.wlan.symbolic.organization`
`user.location.current.wlan.symbolic.country`
	Symbolic address of the user (using WLAN connectivity data)

`....current.mobilecell.geographic.longitude.*`
`...`
`....current.mobilecell.symbolic.name`
`...`
	Current position as determined from cell phone connectivity

`....current.positioning.geographic.longitude.*`
`...`
`....current.positioning.symbolic.name`
`...`
	Current position as determined from dedicated positioning system

`....current.contact.geographic.longitude.*`
`...`
`....current.contact.symbolic.name`
`...`
	Current location from service interaction (known position)

Table 4.3: *Extending the P3P base data schema with location data.* User-submitted location information can either be `geographic` (i.e., coordinates based) or `symbolic`, e.g., a street address. The latter simply reuses the existing P3P postal address schema. As with perception data above, this information can either be self-submitted by the user (`submitted`), or explicitly determined by an external positioning system (`current`). Note that longitude and latitude are further substructured into hours, minutes, and seconds.

```
<POLICY>

  . . .

  <EXTENSION optional="yes">
    <WITHOUT-CONSENT xmlns="http://www.vs.inf.ethz.ch/paws/schemas/updates">
      <extend/>
      <change/>
    </WITHOUT-CONSENT>
  </EXTENSION>

  . . .

</POLICY>
```

Figure 4.3: *Update clause using the* ***WITHOUT-CONSENT****-Extension:* Privacy contracts can optionally contain an extension that describes possible policy updates. Using two elements, `extend` and `change`, data collectors can indicate that they might extend and/or change the current privacy policy without the explicit consent of the user.

As with the perception data described above, location information can either be submitted by the user (e.g., by using a self-positioning system such as GPS) or determined by the service (e.g., by having a sensor register the user at a certain physical location). Explicit positioning by the service must be explicitly described as part of the data collection process, i.e., the corresponding data elements for `location.current` feature four different positioning methods: `wlan` for implicit positioning through wlan access points; `mobilecell` for similar positioning using cell information of a mobile phone; `positioning` for any dedicated positioning infrastructure, e.g., ActiveBat; and `contact` for positioning information acquired through service interaction (e.g., when interacting with a service kiosk). While this might be irrelevant once the user's location is known to the service, it might be nevertheless important information to the user.[6]

4.2.2 Contract Data

A privacy contract consists of a privacy policy (including our ubiquitous computing extensions); an expiration time and date until which it will

[6]Even though it appears that this information could have been much better encapsulated in a single, separate field, e.g., `location.current.method`, the above approach of providing four separate data element blocks is necessary due to the direction of information flow: data elements are for soliciting information from the service user (the data subject) to the service operator (typically the data collector) – a data collector could not use a P3P data element to declare the positioning method in use.

remain valid; the identifiers of the contract partners; and optionally their (digital) signatures.

The policy is a regular P3P policy file that gives a detailed account on the data elements collected (`STATEMENT`), the available dispute resolution mechanism (`DISPUTES-GROUP`), access provisions (`ACCESS`), and optional extensions (`EXTENSION`).

The expiration data of the contract is implicitly given by the corresponding `EXPIRY` element within the P3P policy. This expiration can be given both as an absolute date and time, or relatively to the time the policy has been downloaded. However, from a contract perspective, relative expiration times are difficult to handle, as the download time is not explicitly represented. The current PawS implementation restricts itself to issuing absolute expiration times, though relative times could be handled internally by clients that store the download time as metadata.

The standard `ENTITY` element of P3P can be used to describe both parties – the data collector and the data subject. However, since at the time of issuing the contract no user information is yet available, contracts will most likely only contain explicit information about the data collector. On the data subject side, this should pose no problems. Proving that the data collector really issued the contract only requires a digital signature from the collector – not the identity of the subject. Data collectors on the other hand can optionally associate a link to the data subject's identity with the contract, in case this information is available.

Using the XML digital signature standard [27] (as described in section 3.3.1 above) and its application to P3P [281], policies can optionally be signed by one or both parties. As with the `ENTITY` asymmetry above, signatures will predominantly be used by data collectors to sign their privacy policies. Again, if needed, the same mechanism could also be applied by data subjects to sign a copy of the privacy contract, though this is not implemented in the current PawS prototype.

Optionally, a privacy contract can contain a `WITHOUT-CONSENT` extension, as shown in figure 4.3. This allows data collectors to indicate whether they reserve themselves the right to change or extend their privacy policy without the explicit consent of the user. Obviously, this only makes sense if the data subject in turn has the ability to terminate the contract at any time. If the user agrees to this clause, data collectors are able to extend an existing policy (and thus a contract)

or make minor changes to it without having to exchange a new set of contracts with freshly applied signatures. Last not least, local law will further restrict the applicability of these clauses.

4.2.3 Remote Proxy Access

PawS uses an extension to the regular P3P policy in order to better implement automated access mechanisms. The extension allows a data collector to specify which data fields the data subject can view, edit, or even delete after having submitted his or her personal information to the collector. Additionally, it allows to define the exact access protocol that can then be used by the data subject's privacy proxy (see section 4.3 below) to automatically perform this access, e.g., in order to verify if certain information has been deleted after the expiration date, or if the data subjects wants to update some of his or her personal information.

An example of the `ACCESS-METHODS` element is given in figure 4.4. Using the method elements `UPDATE`, `DELETE`, and `QUERY`, data collectors can declare that the data subject can update, delete, and query the data that was collected on him or her, respectively. Each element in turn contains additional information about the list of data elements this access applies to, as well as the exact access method that a privacy proxy can use to do so.

The regular P3P `EXTENSION` syntax (line 3 in figure 4.4) embeds the additional access information as part of the standard P3P `ACCESS` element (shown in line 1). The link `service.example.org/access` on line 4 provides a human readable description of the access capabilities, which could for example include an email address or telephone number for direct inquiry. Each of the three contained elements – `UPDATE`, `DELETE`, and `QUERY` – then details the access capabilities for the data subject's privacy proxy. Lines 7 through 15 declare that data subjects can update their name, postal address, telephone number, and email address, while lines 16 through 19 specify that all collected information can be deleted at a later time. Lastly, lines 20–32 show that not only the updatable information can be queried for their current contents, but also any dynamically generated data, e.g., shopping patterns or similar secondary information (lines 28–30).[7]

[7]The `p3p:`-prefix is used to reference original P3P syntax elements.

```
<POLICY>

  <ACCESS>
    <all/>

    <EXTENSION optional="yes">

     <ACCESS-METHODS discuri="http://service.example.org/access"
                     xmlns="http://www.vs.inf.ethz.ch/paws/PrivacyContract">

       <UPDATE version="PRO2-1.0" rpc_uri="http://service.example.org/soap/rpc"
                                  service_urn="AccessService"/>
         <p3p:DATA-GROUP>
           <p3p:DATA ref="#user.name"/>
           <p3p:DATA ref="#user.home-info.postal"/>
           <p3p:DATA ref="#user.home-info.telecom.telephone"/>
           <p3p:DATA ref="#user.home-info.online.email"/>
         </p3p:DATA-GROUP>
       </UPDATE>

       <DELETE version="PRO2-1.0" rpc_uri="http://service.example.org/soap/rpc"
                                  service_urn="AccessService"/>
         <all/>
       </DELETE>

       <QUERY version="PRO2-1.0" rpc_uri="http://service.example.org/soap/rpc"
                                 service_urn="AccessService"/>
         <collection/>
         <p3p:DATA-GROUP>
           <p3p:DATA ref="#user.name"/>
           <p3p:DATA ref="#user.home-info.postal"/>
           <p3p:DATA ref="#user.home-info.telecom.telephone"/>
           <p3p:DATA ref="#user.home-info.online.email"/>
           <p3p:DATA ref="#dynamic.miscdata">
             <p3p:CATEGORIES><purchase/><preference/></p3p:CATEGORIES>
           </p3p:DATA>
         </p3p:DATA-GROUP>
       </QUERY>

    </EXTENSION>
  </ACCESS>

  . . .

</POLICY>
```

Figure 4.4: *Detailed access information using the* ***ACCESS-METHODS****-Extension:* Privacy policies in P3P can contain detailed information about possible automated access capabilities the service offers. In the above example, the service at `service.example.org` provides update capabilities to the data subject's name and address; allows subjects to query the data it has on file (including any dynamically generated data); and supports deletion commands for all collected data.

The UPDATE-Element

The UPDATE element (see lines 7–15 in figure 4.4) typically contains a DATA-GROUP element that declares the individual data elements that can be upgraded. The absence of an explicit data element list implies that data subjects have access to all collected data elements. Issuing a separate QUERY command can also give more details about which fields can be updated.

The attributes of the UPDATE element contain information for the data subject's privacy proxy to actually perform the updates. These attributes contain protocol version information (version), the access URL (rpc_uri), and the service identifier (service_urn).[8] Using this information, the data subject's privacy proxy can contact the data collector's proxy and perform updates on the given data elements autonomously and transparantly.

The DELETE-Element

The DELETE element (see lines 16–19 in figure 4.4) uses the same attributes as the UPDATE element, as well as the same DATA-GROUP element to specify which of the collected elements can be deleted at a later time. If no DATA-GROUP element is given, data subject's proxies can assume that all elements can be deleted, though the exact list can again be found out by issuing a QUERY call. A better way to signal the ability to delete all personal information is using the <all/> element (which, incidentially, can also be used in the UPDATE element above). However, whether this possible deletion actually physically erases the personal data in question, or whether it is merely anonymized, is left up to the data collector. More information can again be found behind the URI in the discuri attribute of the encapsulating ACCESS-METHODS element.

The QUERY-Element

The attributes and subelements of the QUERY element are very similar to those of the UDPATE and DELETE elements, nameley the service access attributes (version, rpc_uri, and service_urn) and a DATA-GROUP element enumerating the individual elements whose current value can be queried. Note that additionally available information, such as a shopping profile gathered from individual purchase records, could also

[8] Access URL and service identifier are necessary for performing SOAP function calls. See section 4.3.1.

Service Type	Description
<info/>	The data is collected to provide information to the user, e.g., a timetable application or finding the location of the closest bank. Note that if the service explicitly guides the user to a location (by continuously tracking the current location), the navigation type would need to be used.
<purchase/>	The service is used to sell items other than the service itself (e.g., books or clothing, rather than only a service fee).
<communication/>	The service provides communication capabilities (e.g., mobile phone, pager, or follow-me phone application).
<multimedia/>	Any multimedia application, e.g., live video streaming or audio recording.
<tracking/>	The service tracks the location of the user in order to provide her position to others (e.g., friends or colleagues). If the tracking information is only used for the user herself, the navigation type should be used.
<navigation/>	The service tracks the location of the user in order to guide her to either a specific point of interest, or to provide additional information about the current location (i.e., a tour guide). Note that a tour guide application would also declare the info type as well.
<security/>	The service is in place for security reasons.
optional='yes'	The service is activated after a contract agreement has been reached (this is also the default if no optional attribute is given with the element.
optional='no'	The service is (potentially) always active, no contract agreement will be sought (legal restrictions apply).

Table 4.4: *Describing services using the* SERVICE *extension,* allowing providers to better communicate the type of service they offer, and in turn enabling user's to formulate preferences over a whole range of service types, instead of individual providers. Using the optional attribute, these services are marked as being only active after a contract agreement has been reached or as being mandatory, non-negiotable services, e.g., a security camera in a supermarket.

be listed explicitly here in order to inform the data subject that secondary information is available.

Again, if no DATA-GROUP element is present, or if an explicit <all/> element is given, all collected information should be accessible. The exact list of data elements and the level of access granted can be found out by issuing an empty QUERY call, which prompts the collector's proxy to reply with a list of data elements and their corresponding access abilities.

4.2.4 Ubiquitous Services

P3P uses the `current` purpose to cover a wide variety of purposes, e.g., buying goods online, shipping items, or signing up for a mailing list. Such an approach is possible because of the explicit interaction of a user with a Web page. Clicking on hyperlinks or buttons, and filling out form fields (either manually, or semi-automatically using a browser built-in electronic wallet) implies an awareness of the transaction. In contrast, such transactions are thought to happen automatically in a ubiquitous computing environment, without an explicit involvement of the user, at least for often recurring transactions (e.g., paying a bus fare).

For this purpose we add the `SERVICE` extension, which allows service provider to explicitly describe the type of service they offer. Using the P3P extension mechanism, a `SERVICE` element can be placed within a statement, indicating the type of service that this information is solicited for.[9] This can then be used in a user's preference specifications in order to facilitate more general rules (e.g., give out my location for tourguide applications, but not for purchasing services). A summary of the service-types defined in PawS is given in table 4.4. The optional `mode` attribute can be used to indicate a service's modes of operations, i.e., whether it is *discrete* or *continuous*:

- *Discrete collection services* rely on a single data exchange, e.g., a user registration or an individual order. Even though subsequent interactions with the service can be faciliated by referencing an exiting contract agreement (thus alleviating the need for resubmitting user data), their mode of operation is restricted to individual interactions. Even if a particular descrete collection service is *active*, no user data is collected unless explicitly triggered by the user.

- *Continous collection services* rely on user data covering a specific time span, e.g., a live video feed, or a location tracking over a period of time. Continuous collections imply either a steady stream of explicit data updates on behalf of the user, a service-triggered automated data transfer, or an implicit data collection through sensors under the control of the service provider.

[9] A similar mechanism has since been proposed for the upcoming P3P 1.1 specification, where it is called the *primary purpose extension*.

A continuously operating security camera would thus be declared as `<security mode='continuos'/>` while a single purchase would be indicated using `<purchase mode='discrete'/>`. If no `mode` attribute is given, a service is assumed to be discrete.

In addition, the `SERVICE` element is used to indicate whether the service itself is optional or mandatory (using the `optional='yes'` and `optional='no'` attributes, respectively):

- *Optional services* can be activated by the user through entering into a contract agreement with the service provider. As long as the user does not enter into an agreement, no data is collected.
- *Mandatory services* are continously running and cannot be deactivated by the user. Note that for example a user authentication service that is mandatory for entering a particular building would be considered optional if the user would actively need to enter into a contract agreement before, say, a door would open. In contrast, an automated camera system watching an otherwise unrestricted building entrance would be considered mandatory if it makes recordings of all visitors independent of any contract agreements.

Optional services can additionally be *active*, i.e., the user has agreed to enter an optional service and thus has an existing contract agreement with a particular service provider. This does not necessarily imply a running service, such as an active camera – it only stipulates that the user is authorized to use or configure a particular service. Note that this attribute is not modeled in privacy contracts, but instead is a feature that is managed by the privacy proxies described below.

4.2.5 Summary

PawS privacy contracts are extended P3P policies that take into account the type of interactions present in ubiquitous computing environments (i.e., without user intervention). They include extensions for seamless query, update, and deletion of stored information; descriptions for ubiquitous computing services that facilitate preferences across a variety of different service providers; and an extended set of data elements for location and perception data.

Figure 4.5 shows a high-level summary of a PawS privacy contract, illustrating how the various components – P3P policy, statements, ac-

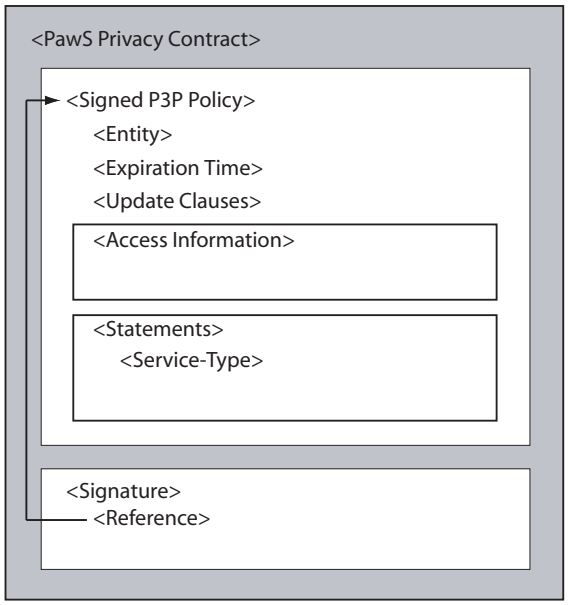


Figure 4.5: *PawS privacy contract.* Privacy contracts consist of a regular, optionally signed P3P policy with a number of PawS specific extensions, such as access methods, service type information, and additional data schemas particular to ubiquitous computing environments.

cess method, service type, and digital signature – are arranged. The next section will describe how these contracts are used in PawS, i.e., privacy *proxies* for both services and users will exchange them in order to enable seamless service usage while maintaining the user's privacy (or at least make it transparent when it has been violated).

4.3 Privacy Proxies

The privacy proxy is the main architectural element in PawS.[10] It is used both on the user and the service side, and is responsible for the seamless exchange of service policies and user submitted data. Privacy proxies support two basic modes of operation:

1. *Requesting, serving, and agreeing on privacy policies:* Data collections in privacy aware ubiquitous computing environments are tagged with the URL of the corresponding privacy policy,[11] typically hosted by the data collector's service proxy[12] and requested by a user's privacy proxy.

[10]Privacy proxies were developed as part of the diploma thesis of Mark Stäheli [325].
[11]See section 4.4 for a description of the dissemination mechanism.
[12]In practice, any Web server can be used for serving policies, though a tight integration with the service proxy is desirable. As we implemented our service proxy on top of an Apache Web server, our proxies can easily provide simple page serving as well.

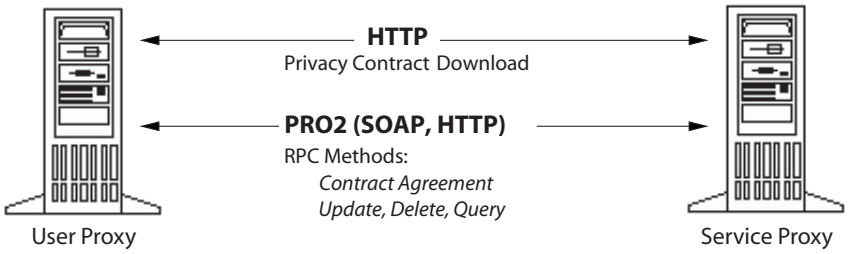


Figure 4.6: *PRO2 overview.* Privacy proxies use HTTP to exchange privacy policies and contracts, and PRO2 (which is based on SOAP over HTTP) for supporting contract agreements and remote access.

2. *Data access interface:* Besides the actual data submission from the user proxy to the service proxy, service proxies may optionally support direct query, update, and deletion access to any stored user data. Similarly, user proxies might optionally allow services to update or delete existing contracts, as well as directly query (parts of) the user repository for any updates.

These two aspects will be described in more detail in the next two sections, where we will first outline the proxy protocol (PRO2), followed by a detailed description of our *contract agreements* and how proxies can use the informatio contained in them to facilitate both contract and data management.

4.3.1 The Privacy Proxy Protocol (PRO2)

Figure 4.6 gives a summary of the proxy tasks and the role of PRO2. It shows the two phases in a user and service proxy interaction, which begins with a regular HTTP interaction where the user client requests the privacy policy of a particular service from a URI situated at the service proxy. Only after the user proxy has successfully downloaded the XML policy file do subsequent interactions use PRO2.

Figure 4.7 illustrates an example: a user proxy obtains a link to a service's privacy policy at `service.example.com/servlet/contract` and uses a standard HTTP GET-request as shown in subfigure (a) to request it. The corresponding servlet on the service proxy replies with the XML policy file (see subfigure 4.7.b) and includes PRO2 specific HTTP headers in its reply (lines 3-5): `PRO2_contract_id`, `PRO2_rpc_uri`, and `PRO2_service_urn`. The `rpc_uri` and the `service_urn` together form a complete URI, while the `contract_id` serves as an identifier for the subsequent interaction between user proxy and server proxy.

```
GET /servlet/contract HTTP/1.1
Host: service.example.com
User-Agent: PRO2-Proxy
```

(a) User proxy request

```
HTTP/1.1 200 OK
Date: Thu, 28 Feb 2004 17:55:07 GMT
PRO2_contract_id: 280201-23855671
PRO2_rpc_uri: http://service.example.com/soap/servlet/rpc
PRO2_service_urn: ContractService
Content-Length: 3855
Content-Type: text/xml

<?xml version="1.0"?>
<POLICIES xmlns="http://www.w3.org/2001/09/P3Pv1">
  <EXPIRY max-age="Sun, 30 Jun 2002 23:59:59 GMT"/>
  <POLICY discuri="http://service.example.com/P3P/PrivacyContract.html"
          name="LocationService">
     ...
  </POLICY>
</POLICIES>
```

(b) Service proxy reply

Figure 4.7: *Privacy contract download.* The first phase of any proxy interaction is the privacy contract download. It uses regular HTTP requests (a), while the HTTP replies from the service proxy contain additional headers suitable for setting up the subsequent PRO2 interaction (b).

Before we give further details on the PRO2 interaction, we will briefly describe the HTTP and SOAP protocols, on which PRO2 is based upon.

Transport Layers: HTTP and SOAP

HTTP is the standard protocol for the Web, and is typically situated on top of TCP/IP. HTTP is a request/response protocol, meaning that a client sends a request to a server in the form of a request method, URI, and protocol version, followed by optional request modifiers, client information, and possibly body content [116]. An example can be seen in figure 4.7.[13] In its reply, the server indicates the protocol version, content type, and length, followed by the actual resource.

SOAP – the *Simple Object Access Protocol* – is an XML-based protocol to exchange "structured and typed information between peers in a decentralized, distributed environment" [245]. It typically runs over HTTP and is well suited to represent *remote procedure calls* (RPCs), i.e., function calls outside the calling procedure's address space, either on the same machine or on different systems connected by a network.[14] Specifically, the two parts of the SOAP specification, i.e., the messaging framework [149] and the adjunct specification [150], define:

1. *A message encoding format,* consisting of a SOAP *envelope* that holds an optional SOAP *header* and a mandatory SOAP *body*. Both header and body carry application specific data (its contents are thus not defined in SOAP), with the body containing the "end-to-end information" conveyed in the SOAP message and the header carrying "control" information that is not considered application payload.

2. *A message processing model,* which describes the actions a SOAP node (i.e., a computer capable of receiving and processing SOAP messages) must take upon receiving a SOAP message. This includes checking the message for syntactic correctness and parsing all SOAP specific attributes, which indicate a) how a specific node should handle the message, b) what parts of the message a node must understand, and c) how parts of the header are to be relayed to potential follow-up nodes.

[13] The `GET` method is used to retrieve the particular resource indicated by the URI from the server. The `POST` method in contrast allows clients to submit additional information in the request body, see figure 4.8 for an example.

[14] Also often called *remote message invocation* (RMI) in the context of object oriented programming languages.

```
POST /stoxx/cgi-bin/stockquote HTTP/1.1
Host: www.stoxx.example.com
Content-Type: text/xml
Content-Length: 415

<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/"
              env:encodingStyle="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Header>
    <tx:Transaction xmlns:tx="http://www.stoxx.example.com/ticketing"
                    env:mustUnderstand="1">
      jT56Hbvdo8lsVReik3q2LX7q
    </tx:Transaction>
  </env:Header>
  <env:Body>
    <method:GetStockQuote xmlns:meth="http://www.stoxx.example.com/methods">
      <Symbol>SAir</Symbol>
    </method:GetStockQuote>
  </env:Body>
</env:Envelope>
```

(a) SOAP RPC request

```
HTTP/1.1 200 OK
Content-Type: text/xml
Content-Length: 297

<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/"
              env:encodingStyle="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Body>
    <method:GetStockQuoteResponse xmlns:meth="http://www.stoxx.example.com/methods">
      <Price>17.50</Price>
    </method:GetStockQuoteResponse>
  </env:Body>
</env:Envelope>
```

(b) SOAP RPC response

Figure 4.8: *Example of a Remote procedure call with SOAP.* Using the HTTP protocol, a RPC request for a particular stock quote is sent via an HTTP-POST method to a SOAP node (a). The response is sent back via the regular HTTP protocol as well (b).

3. *An optional data model,* that may be used by applications to map non-XML data to an XML representation.

4. *A protocol binding framework,* which can be used to exchange SOAP messages over a variety of protocols, most notably HTTP.

5. *A specific representation for RPCs,* that defines the format of RPC invocation and response messages in a SOAP body, as well as a list of standardized error codes, called *RPC faults*.

Figure 4.8 gives an example of a simple RPC interaction (request and response) encoded in SOAP and sent over HTTP.

4.3.2 Contract Agreements

After having downloaded the privacy contract from the service proxy, a user proxy compares it with the preferences of the user. Just like in P3P, there is no explicit negotiation phase in PawS – it is "take it or leave it:" services disclose their offers and corresponding data collection requirements, and users can (if possible)[15] decide whether to use the service. We have not incorporated an explicit automated preference mechanism in PawS, as the focus was primarily on the overall infrastructure. However, there exist a number of preference formulation languages [12, 80], as well as implementations [46] for such a task, and future development of PawS will focus more prominently on usability aspects (see future work in section 7.2).

In the following we thus assume an acceptable privacy contract has been sent by the service proxy, and that the user proxy – either by automatic means or through manual user selection – wants to enter into an agreement. PRO2 differntiates between three cases:

1. *New contract:* The user does not have an existing contract with the particular service provider. Besides sending the required data (and possibly some optional data) under the contract's `contract_id`, the user proxy optionally communicates its potential remote access capabilities, e.g., for allowing the service provider to update its policy at a later time (see section 4.2.3 above).

2. *Contract update with new data:* The user already has a contract with this particular service provider, though she wants to switch to the new one, which also involves sending additional data (or updating existing data that the service has on file from her). This involves sending the *old* `contract_id` in addition to the new one, as well as submitting any additional personal data that is necessary.

3. `Contract update with existing data:` The user already has a previous contract (albeit with slightly different parameters), and no new data is necessary. In this case, only the new and old `contact_id`'s need to be sent.

These three cases map onto three different `contract`-methods with different calling signatures and return values, as shown in figure 4.9.

[15] In many instances, data is (and can be) collected without the user's explicit consent, e.g., CCTV cameras in supermarkets or public buildings.

```
int contract (String contract_id, String xml_userdata,
                                  String xml_proxydata)
```

(a) New contract

```
int contract (String old_contract_id, String new_contract_id,
              String xml_userdata, String xml_proxydata)
```

(b) Contract update with new data

```
int contract (String old_contract_id, String new_contract_id)
```

(c) Contract update with existing data

Figure 4.9: *PRO2* `contract`*-methods.* Depending on whether the user already has a previous contract with the service provider, and whether new data is being collected, three different calling signatures and return values are used in PRO2.

The `xml_userdata`-element contains the actual user data that is sent to the service provider. It uses a `USER-DATA` element which encapsulates the individual fields according to the base data schema. An example of such a transmission could be

```
<USER-DATA>
  <STATEMENT>
    <PURPOSE><current/><financial/></PURPOSE>
    <RECIPIENT><ours/></RECIPIENT>
    <DATA-GROUP>
      <DATA ref="#user.name.given">John</DATA>
      <DATA ref="#user.name.family">Doe</DATA>
      ...
    </DATA-GROUP>
  </STATEMENT>
</USER-DATA>
```

Since the P3P specification allows policies to contain not only optional data elements, but also optional purposes or recipients,[16] it is not sufficient to simply submit the data elements in case of a new contract agreement. The submission must also indicate both the intended recipients and the intended valid purposes of the user's data disclosure. If the `xml_userdata` element contains no `PURPOSE` or `RECIPIENT` element,

[16] Both the subelements to the `PURPOSE` and the `RECIPIENT` element can carry an optional `required` keyword that can assume either the value `opt-in` or `opt-out`. See figure 4.19 or the latest P3P specification [79] for an example.

```
<PROXY-DATA>
  <ACCESS-METHODS>
    <UPDATE version="PRO2-1.0"
            rpc_uri="http://www.myproxy.example.org/soap/rpc"
            service_urn="ServiceAccess"/>
    <DELETE version="PRO2-1.0"
            rpc_uri="http://www.myproxy.example.org/soap/rpc"
            service_urn="ServiceAccess"/>
    <QUERY version="PRO2-1.0"
            rpc_uri="http://www.myproxy.example.org/soap/rpc"
            service_urn="ServiceAccess"/>
  </ACCESS-METHODS>
</PROXY-DATA>
```

Figure 4.10: *Example of a PRO2* `xml_proxydata`*-element.* User proxies can explicitly offer service proxies remote access capabilities, e.g., for updating individual privacy contract parameters or for verifying the data they have on file.

all elements of the original policy carrying a `required="opt-out"` attribute are assumed to be agreed to by the user, while elements that carry a `required="opt-in"` attribute are not agreed to. In case the user specifies a purpose or recipient that was not part of the original policy, a `data not accepted` error is returned by the service proxy. Similarly, if the user leaves out a required (neither opt-in nor opt-out) purpose or recipient, the server responds with a `data not complete` error.[17]

The `xml_proxydata`-element contains the (optional) remote access capabilities of the user proxy. As described in section 4.2.3 above, user proxies can allow service proxies to query, update, and potentially delete stored privacy contracts. Service providers might for example use this mechanism to verify a user's home address one last time before shipping a product, or to renew an otherwise expiring privacy contract, or to demonstrate its data deletion fulfillment by deleting the corresponding privacy contract it was collected under.[18] Figure 4.10 shows an example of a `xml_proxydata` element.

Note that while PawS currently supports digital signatures for privacy contract, user proxy replies are not signed. In order to extend PawS to handle user signatures as well (e.g., for allowing data collectors to prove user consent in case of disputes), privacy contracts would need to be extended to be able to carry a corresponding "user-signature-required"-

[17] See table 4.5 for all possible return values.

[18] Note that upon receiving a delete request, a user proxy would typically *archive* a particular privacy contract, rather than deleting all traces of it.

0	`ok`
The contract was successfully created or updated.	
1	`no such contract_id`
The referenced contract id could not be found.	
2	`call not allowed or not supported`
The user called an unsupported method.	
3	`data not accepted`
The service proxy did not accept the submitted data due to malformed elements (e.g., wrong email address format).	
4	`data not complete`
The service proxy did not accept the submitted data due to missing (but required) elements.	
5	`old contract not found`
The service could not find the referenced *old* contract.	
6	`contract update not possible`
The user is not allowed to update an existing old contract with the reference new contract.	

Table 4.5: *Return values for the PRO2* `contract`*-methods.*

field. User proxy replies to the initial privacy contract download would then feature an enveloping or enveloped[19] XML signature.

Should the privacy contract be acceptable, the user proxy thus creates a SOAP call using the corresponding method interface from figure 4.9 and sends it to the SOAP RPC URI given in the HTTP header (see lines 3–5 in figure 4.7 on page 137). Table 4.5 lists the possible return values of this call, which are sent back from the service proxy. Note that this assumes that user proxies are aware of the semantics of the base data elements, i.e., that some of the PawS base data schema extensions do not actually describe data that is to be collected directly from the user, but instead through sensor operated by or accessible to the data collector's service proxy, such as the entire `perception.current` data set.

Once data has been sent from the user proxy to the service proxy, i.e., a contract agreement has been reached, the user is ready to use the offered service (e.g., a follow-me telephone service, or a single ride on a public train). Both the service proxy and the user proxy have now a record of the transaction in storage (i.e., a copy of the privacy contract) and can monitor both expiration time (i.e., when the collected data must be deleted) and usage restrictions (i.e., complying with the

[19] See section 3.3.1.

given purpose and recipient information on the data collector's side). Additionally, as long as a contract is active, both user proxy and service proxy can use the advertised access methods to query the remote storage repository to ensure data integrity or verify contract status. These access methods will be described in the following two sections.

Remote Repository Access (User Accesses Service Data)

The ACCESS-METHODS extension described in section 4.2.3 above allows data collectors to provide data subjects with a direct link to the personal information they have stored about them. Figure 4.11 shows the calling signatures for the three types of user access methods.

For triggering an update of personal data stored at the service proxy, the user proxy simply sends the *contract-id* under which the data was collected (or the contract ID of the latest policy update) together with an XML representation of the new data elements within an UPDATE element:

```
<UPDATE>
  <STATEMENT>
    <PURPOSE><marketing/></PURPOSE>
    <DATA-GROUP>
      <DATA ref="#user.employer">ETH Zurich</DATA>
      <DATA ref="#user.department">Computer Science</DATA>
    </DATA-GROUP>
  </STATEMENT>
</UPDATE>
```

The user can specify PURPOSE and RECIPIENT elements to update the list of valid purposes and recipients of his or her personal data. If such purpose or recipient updates appear without a DATA-GROUP element, they are supposed to apply to all collected user elements; otherwise the purpose update request only applies to the given elements.[20] Empty DATA elements do not delete the corresponding values – this must be done using the delete method.

A similar calling structure is used for deleting data from the repository, with the exception that the elements that should be deleted are inside a DELETE element and must not carry any actual data. Also, no PURPOSE or RECIPIENT elements are allowed (the contents start directly with the DATA-GROUP element, not with a STATEMENT element):[21]

[20] Note that if only the purpose or recipient of a specific element should be updated, but not its data, an empty <DATA ref="#..."/> element should be used.

[21] If a specific purpose or recipient should be deleted, the update method must be used.

```
int update (String contract_id, String xml_update)
```

(a) Data update

```
int delete (String contract_id, String xml_delete)
```

(b) Data deletion

```
QueryResponse query (String contract_id, String xml_query)
```

(c) Data query

Figure 4.11: *PRO2* `user-access-methods`. If a service supports extend access capabilities, the user proxy can contact the service proxy at any time in order to query the data the service has on file on her, update it if necessary, or even delete some or all of the data.

```
<DELETE>
  <DATA-GROUP>
    <DATA ref="#user.home-info.online.email"/>
    <DATA ref="#user.home-info.telecom.mobile"/>
    ...
  </DATA-GROUP>
</DELETE>
```

In addition, the `<all/>` element can be used to request deletion of all personal data, which implies deletion of the entire contract. For both delete and update requests, the service proxy replies with a numeric return value that indicates the success or (partial) failure of the requested operation (see table 4.6).

The `query`-method uses a similar calling signature as the `delete`-method, i.e., a `contract_id` and a list of XML elements to be queried, given inside a `QUERY` element. In particular, the `QUERY` element supports single data element entries (compare with the `DELETE` example above) for querying a set of specific elements, the `<all/>` element for receiving a list of all stored (and accessible) elements, and the special `<collection/>` element for querying data collection parameters. An example for such a reply is given in figure 4.12.a: The service has the user's name and gender on file and uses it for the purpose of `current` and `marketing`, with only itself (`ours`) as valid recipients. While both the purpose and recipient information cannot be changed, the service supports updates on the user's name and jobtitle, and deletions on both

0	ok
Data has been successfully update or deleted.	
1	no such contract
The service provider could not find a contract under the referenced ID. No updates or deletions have been performed.	
2	call not allowed or not supported
The service proxy does not support or allow the requested operation. The corresponding policy should be referenced for a list of supported operations. In case of disputes, the information given in the DISPUTES-GROUP element should be perused.	
3	data not accepted
The submitted XML data could not be properly parsed.	
4	updates/deletes only partially accepted
Some of the requested updates or deletions have failed due to access restrictions or, in case of updates, format errors (e.g., malformed email addresses). Use the query mechanism to find out which elements have not been updated or deleted.	

Table 4.6: *Return values for the PRO2* `update` *and* `delete` *methods.*

optional data elements (i.e., jobtitle and gender).

The return value for the `query`-method is a compound value (denoted `QueryResponse` in figure 4.11.c), consisting of an error code (`int`) and a `String` containing the values of the queried elements, returned inside an `ANSWER` element. An example for such a reply is shown in 4.12.b.

The detailed description of possible error codes is given in table 4.7. Note that in most error cases, the `xml_answer` string will be empty.

Contract Updates

User proxies can optionally offer access functionality similar to the above service proxy access methods. These methods are not part of the privacy contract, but are transmitted separately by the user proxy as part of its reply using the `xml_proxydata` parameter of the `contract`-method (see section 4.3.2). Similar to the previous service proxy methods, user proxies can support queries, updates, and deletes. However, instead of operating on user data, user proxy access methods refer to the contract instead (even though the calling signature is exactly the same as for the service proxy access methods, see figure 4.11).

The `update`-method allows service proxies to notify the user proxy of updates to its privacy policy. This can either be an optional up-

```
<ANSWER>
  <STATEMENT>
    <PURPOSE>
      <current/>
      <marketing state="yes" required="opt-in" update="no"/>
    </PURPOSE>
    <RECIPIENT>
      <ours/>
      <others state="no" required="opt-in" update="no"/>
    </RECIPIENT>
    <DATA-GROUP>
      <DATA ref="#user.name.given" state="complete" optional="no"
            update="yes" delete="no" query="yes"/>
      <DATA ref="#user.name.familiy" state="complete" optional="no"
            update="yes" delete="no" query="yes"/>
      <DATA ref="#user.gender" state="complete" optional="yes"
            update="no" delete="yes" query="yes"/>
      <DATA ref="#user.jobtitle" state="empty" optional="yes"
            update="yes" delete="yes" query="yes"/>
    </DATA-GROUP>
  </STATEMENT>
</ANSWER>
```

(a) Collection query

```
<ANSWER>
  <STATEMENT>
    <PURPOSE>
      <current/>
      <marketing/>
    </PURPOSE>
    <RECIPIENT>
      <ours/>
    </RECIPIENT>
    <DATA-GROUP>
      <DATA ref="user.home-info.online.email">
        example@example.com
      </DATA>
      <DATA ref="user.home-info.telecom.mobile">
        +99 (123) 4567-890
      </DATA>
    </DATA-GROUP>
  </STATEMENT>
</ANSWER>
```

(b) Data query

Figure 4.12: *Examples of PRO2 user queries.* User proxies can query service proxies for the current state of their repository, i.e., what data elements are present, under what purpose and recipient these have been collected, and what access methods the service proxy supports for each of them (a), or query the actual data on file (b).

0	ok
The query was successfully processed.	
1	no such contract
The service provider could not find a contract under the referenced ID (xml_answer is empty).	
2	call not allowed or not supported
The service proxy does not support or allow the requested operation (xml_answer is empty).	
3	data not accepted
The submitted XML data could not be properly parsed (xml_answer is empty).	
4	queries only partially possible
Some of the queried elements could not be returned, either due to access restrictions or because the elements have not been stored. use the <collection/> query to find out which elements are stored and accessible.	

Table 4.7: *Return values for the PRO2* `query`*-method.*

date, e.g., a request for extension of an existing contract, or a notice of an automated extension or update according to the WITHOUT-CONSENT extension that was present in the original contract (see section 4.2.2 on page 128). The contract_id element identifies the contract to be updated, while the xml_update string contains either a NEW-CONTRACT or a EXTEND-CONTRACT element inside an UPDATE element.

Using the NEW-CONTRACT element, service proxies can indicate that they would like to update the existing contract with an updated version. The contract_uri attribute contains a link to the new contract, while the DEFAULT element describes the consequences if the user proxy does not explicitly accept or deny this request until the expiration time indicated in the EXPIRY element. The default consequences are:

1. delete: The existing contract will be deleted and no new agreement is being made.

2. retain: The existing contract will continue to apply until its expiration time.

3. update: The existing contract will be updated with the new contract.

It of course depends on local legislation what kinds of defaults are possible for contract updates. In most legislations, only a delete or

```
<UPDATE>
  <NEW-CONTRACT contract_uri="http://service.example.com/new/PrivacyContract">
  <DEFAULT>delete</DEFAULT>
  <EXPIRY date="Tue, 31 Dec 2004 23:59:59 GMT"/>
  <INFO> Due to new EU regulation 04/12EC, our policy now contains
    an explicit link to the legal framework they are bound by. </INFO>
</UPDATE>
```

(a) New contract update

```
<UPDATE>
  <EXTEND-CONTRACT>
    <EXPIRY date="Tue, 31 Dec 2005 23:59:59 GMT"/>
  </EXTEND-CONTRACT>
  <DEFAULT>delete</DEFAULT>
  <EXPIRY date="Tue, 31 Dec 2004 23:59:59 GMT"/>
  <INFO> We automatically extended the contract validity for another
    year. Thank you for your trust in our services. </INFO>
</UPDATE>
```

(b) Contract extension

Figure 4.13: *Examples of PRO2 user proxy contract updates.* Similar to the access methods of service proxies, user proxies can offer services to update, query, and delete the contracts the user proxy has agreed to with a particular service.

`retain` would probably be allowed, unless the existing contract specifically contains a `WITHOUT-CONSENT`-element. If no `DEFAULT`-element is given, a `delete` consequence must be assumed. Similarly, if no `EXPIRY`-element is present, the expiration date of the existing (referenced) contract must be assumed.

Alternatively, service proxies can request an extension of an existing contract using the `EXTEND-CONTRACT`-element. It again uses the `DEFAULT`-element to indicate the default consequence and the `EXPIRY`-element to indicate the time until the service proxy expects a reply. If the `DEFAULT`-element is left out, a `delete` consequence is assumed (i.e., the contract is not extended but expires), while a missing `EXPIRY`-element requires both sides to assume the expiration time of the referenced contract until the user proxy should reply.

The individual return values of the method are given in table 4.8. Note that these are returned *from* the user proxy *to* the service proxy. Also, the immediate return value of the method call does not include the user proxy's reply yet, only an indication whether the contract update request properly reached the user proxy. For this the user proxy can, at any time within the indicated expiration time, use the `contract`-method call described in section 4.3.2 above, using either the contract

0	ok
The user proxy has successfully received the service proxy's update request.	
1	no such contract
The user proxy could not find a contract with the ID indicated by the service.	
2	call not allowed or not supported
The user proxy does not support updates.	
3	data not accepted
The data in xml_update could not be parsed.	

Table 4.8: *Return values for the user proxy's* `update`*-method.*

0	ok
The user proxy has successfully received the service proxy's update request and will delete the contract at the specified expiration time.	
1	no such contract
The user proxy could not find a contract with the ID indicated by the service.	
2	call not allowed or not supported
The user proxy does not support explicitly scheduled deletions.	
3	data not accepted
The data in xml_update could not be parsed.	

Table 4.9: *Return values for the user proxy's* `delete`*-method.*

update method call that includes additional user data submissions, or the one without, depending on the actual newly proposed contract.

The `delete`-method allows service proxies to notify the user of a scheduled or unscheduled contract expiration.[22] While the required `contract_id` parameter contains the corresponding contract that is supposed to expire, the optional `xml_delete` can indicate a human-readable description of the expiration reason, as well as an optional expiration time.[23] An example of such a delete message might look like the following:

```
<DELETE>
  <EXPIRY date="Tue, 31 Dec 2004 23:59:59 GMT"/>
  <INFO discuri="http://service.example.com/info/service-ended.html">
    We will discontinue our service at the end of the year.
  </INFO>
</DELETE>
```

[22] See figure see figure 4.11.b on page 145 for its calling signature.
[23] A missing expiration time assumes the contract's scheduled expiration time.

Table 4.9 lists the possible return values for this call. Note that user proxies will typically archive an expired contract rather than delete it.

The user proxy's `query`-method allows service proxies to individually query user data elements, e.g., for verifying the integrity of previously submitted data such as addresses, but also for repeatedly requesting dynamic data such as user-submitted location information. The calling signature contains the `contract_id` parameter for indicating the privacy contract under which the data is collected, while the `xml_query` parameter contains the actual elements that the service wants to query (inside a `QUERY`-element), with an optional `INFO`-element that can be used to give a brief explanation and/or link to a human-readable Web page with details. A typical `xml_query` string would thus look like this:

```
<QUERY>
  <DATA-GROUP>
      <DATA ref="#user.location.submitted.symbolic.city"/>
      <DATA ref="#user.location.submitted.symbolic.street"/>
  </DATA-GROUP>
  <INFO discuri="http://service.example.com/locate/description.html">
    We need your current location in order to provide you with
    updated information as part of your WHERE_AM_I(tm)-subscription.
  </INFO>
</QUERY>
```

User proxies have two ways of responding to such a request: either immediately through the method's return value (i.e., as part of the `QueryResponse`, see the identical method signature of the corresponding service method in figure 4.11 on page 145 above), or at a later time using the service proxy's `update` method described in section 4.3.2. An example for an immediate reply would look like this:

```
<ANSWER>
  <DATA-GROUP>
    <DATA ref="#user.location.submitted.symbolic.city">Zuerich</DATA>
    <DATA ref="#user.location.submitted.symbolic.street">Paradeplatz</DATA>
  </DATA-GROUP>
</ANSWER>
```

For delayed replies or in case of an error, the `xml_answer` string of the `QueryReply` return value remains empty. The user proxy can use the error codes described in table 4.10 to indicate whether it replies immediately, or asynchronously using the server's access methods.

0	`ok`
The user proxy has successfully received the service proxy's query – the required data can be found in the `xml_answer` return value.	
1	`no such contract`
The user proxy could not find a contract with the ID indicated by the service.	
2	`call not allowed or not supported`
The user proxy does not support queries.	
3	`data not accepted`
The data in `xml_update` could not be parsed.	
4	`no such data`
The requested data is not part of the referenced contract agreement.	
5	`answer delayed`
The requested information will be submitted using the service proxy's `update`-method. `xml_answer` remains empty.	

Table 4.10: *Return values for the user proxy's* `query`*-method.*

4.3.3 Proxy Security

A data exchange between two privacy proxies potentially includes the transfer of more or less sensitive personal information. In order for the service proxy to be able to ensure the proper handling of user data in accordance with the agreed upon privacy contract, this data must be protected from third parties during transfer and storage. While the storage aspects will be discussed in our privacy database section (see section 4.5), we will look at the security of the data transfer between user proxy and service proxy in the following paragraphs. We will begin with briefly summarizing common security requirements of data transfers over public networks, and then demonstrate how we can use established technical privacy mechanisms – SSL, digital signatures, and mix-networks, which we introduced in sections 3.3.1 and 3.3.2 – to safeguard PRO2 communications.

Network Security

The ISO/OSI security architecture [177] defines the following five services in order to provide network security:

- *Data confidentiality:* Messages should be protected from unauthorized disclosure.
- *Data integrity:* The receiver of a message should be able to verify

that the message has not been modified in transit. Likewise, an intruder should not be able to substitute a legitimate message with a false one, or mask a newly created message as a legitimate message (e.g., replay attack).

- *Data origin authentication:* The receiver of a message should be able to ascertain the origin of the message. An intruder should not be able to masquerade as someone else.
- *Peer-entity authentication:* Similarly, the sender of a message should be able to validate the recipient of the message.
- *Non-repudiation:* Senders should not be able to falsely deny later that they sent a particular message.[24]

A public-key infrastructure can be used to cover the first four of these security requirements – confidentiality, integrity, sender authentication, and peer authentication [7]. The SSL protocol introduced in section 3.3.1 is such a public-key based connection encryption tool and can provide all of these features. However, since most users do not have identity certificates, SLL typically only ensures the identity of the Web server through the required use of server certificates, even though client authentication is possible. Just as PawS does not yet support user-signed privacy contracts, it does not use any user certifcates. Should these be in widespread use, it could easily be extended to authenticate the user as part of the proxy protocol as well.

What SLL does not provide is non-repudiation: either party can still claim that it did not send a message that the other did receive. As we have seen in section 3.3.1 above, digital signatures can provide such a service: by "signing" a certain message with the secret key,[25] the authenticity of a message can be proven using the corresponding public key.[26] Since the actual messages sent in PawS are using the SOAP message format, we rely on the SOAP-DSIG initiative [50] that defines a standard way of using the XML digital signature syntax to sign SOAP messages.

[24] ISO 7498-2 actually requires non-repudiation for sender and receiver, using the terms *proof of origin* and *proof of delivery*. While it would be possible to provide both sender and receiver non-repudiation by introducing additional acknowledge messages, PawS only employs sender non-repudiation.

[25] In practice, only the *message digest*, a much shorter hash of the original message, is signed, i.e., encrypted with the secret key.

[26] Obviously, what is really proven is only that the message sender was in posession of the private key – if the private key is stolen or publicly disclosed, the authenticity of subsequent message is limited.

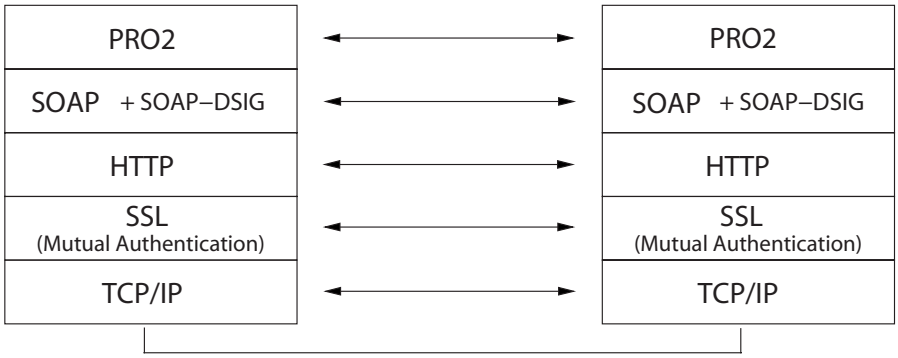


Figure 4.14: *PRO2 secure protocol stack.* Using secure communication and authentication via SSL running over TCP/IP, proxies can exchange signed SOAP messages (using SOAP-DSIG) over HTTP.

Using both SSL and SOAP-DSIG digital signatures, PawS privacy proxies can thus exchange SOAP messages in a secure fashion, i.e., ensuring data confidentiality, integrity, authenticity, and non-repudiation. Figure 4.14 depicts the protocol stack used in PRO2 applications:

1. The defined PRO2-RPC-method is encoded using the SOAP RPC encoding.
2. A digital signature is added to the SOAP message using SOAP-DSIG, a SOAP compatible XML digital signature.
3. The entire SOAP message (including the signature) is used in the body of an HTTP request (or an HTTP reply), i.e., HTTP headers are prepended.
4. After having established an SSL connection to the recipient, the plain-text HTTP data is encrypted according to the agreed-upon encryption algorithm and key.
5. The encrypted data is sent via TCP/IP. Even if individual packets are exchanged, the SSL layer will ensure that end-to-end confidentiality and integrity is preserved.

Network Anonymity

Equally important for our privacy architecture is the option of exchanging such information anonymously, i.e., the actual communication over the network should not result in the disclosure of the user's IP address, which might allow the service provider or a third party to infer the user's full identity.[27] While this might seem contradictory given

[27] Compare with section 3.3.2.

our quest for authentication and non-repudiation above, there are two important reasons for such a requirement:

1. Not all data exchanges between a service and a user require the user's identity. An indoor navigation service does not require a user's name or phone number in order to dynamically guide her to a specific location – it might only require valid payment, either in the form of a credit card, or using anonymous e-cash [61].

2. In order to send SOAP messages back and forth between two privacy proxies, the IP addresses of the respective recipients must be known. As these are typically logged (or can potentially be logged) at various points along the network route, as well as at the individual proxies themselves, they constitute a separate data collection independent of the application-level data exchange between user and service. This complicates data management for the service proxy operator, as such log data must not only be explicitly declared in the privacy contract, but also incorporated into the data storage concept based on PawDB.

In contrast to the confidentiality and integrity requirements described above, however, which needed to be taken into account at the architectural level, we can rely on external tools such as JAP[28] to provide user proxies with the ability to connect to a service proxy anonymously.

4.3.4 Implementation

We implemented both user and service proxy in our PawS prototype on top of a regular Apache Tomcat installation.[29] Tomcat is an *application server*, i.e., it provides applications to thin clients (Web browsers) that are being run and managed by the server. It is the official reference implementation of the corresponding Java application server framework from Sun, supporting *Java Servlets*[30] and *JavaServer Pages*.[31] The individual proxy modules are Java servlets that are triggered by HTTP requests, just as a Web server is triggered to return (often static) Web pages to Web browsers.

[28] See `anon.inf.tu-dresden.de`
[29] See `jakarta.apache.org/tomcat`
[30] Java Servlets are Java programs that can be invoked over HTTP, e.g., for providing dynamic Web pages. See `java.sun.com/products/servlet/index.jsp`
[31] See `java.sun.com/products/jsp/`

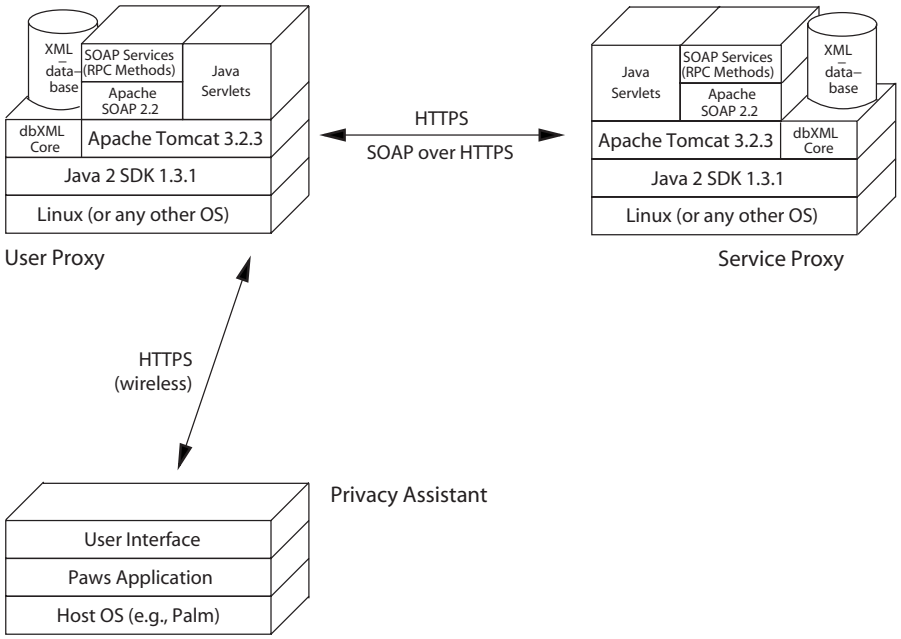


Figure 4.15: *Proxy technology overview.* Privacy proxies run on Apache Tomcat application servers, using Apache SOAP for communication and a small XML database for storage.

The *Apache SOAP* project[32] is an open-source, Java-based implementation of the SOAP 1.2 specification [149] which runs on top of an Apache Tomcat installation. Apache SOAP supports both SOAP *messaging* and *remote procedure calls* via SOAP, i.e., it enables the Tomcat server to support SOAP-RPC interfaces.

All application data (i.e., privacy contracts, agreements, etc.) is managed using the native XML database *Apache Xindice*, which also runs under the Tomcat application server and provides a simple interface to saving and retrieving XML documents. Xindice implements the *XML:DB* programming interface, a standardized API for XML databases.[33] A separate XML parser package, *Xerces*,[34] allows our Java servlets to fully parse and process XML data.

Figure 4.15 shows how the individual technologies are used in our proxy implementation. The principal architecture is shown in figure 4.16, illustrating how the individual system parts make up a user and service proxy.

[32]See `ws.apache.org/soap/`
[33]See `xmldb-org.sourceforge.net/`
[34]See `xml.apache.org/`

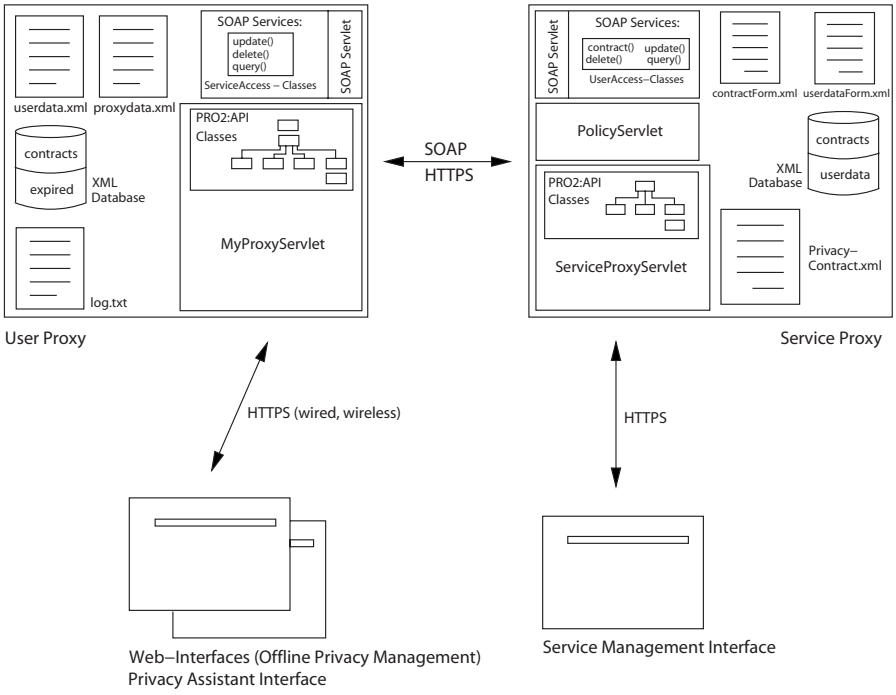


Figure 4.16: *Proxy architecture overview.* Both user and server proxy share a significant amount of code. They only differ in the kinds of SOAP services they offer.

4.3.5 Summary

Privacy proxies are the core element of our PawS architecture. They use extended P3P policies, called *privacy contracts*, in order to communicate both the terms of data collections and the means to access this information at a later time. The PRO2 protocol governs the exchange of such messages, using SOAP over secure SLL connections and incorporating digital signatures for integrity and non-repudiation. Using anonymizing services such as mix networks or anonymizing proxies, users can optionally hide their IP addresses for improved privacy at the network level.

What has not been discussed yet is how these policies can be communicated to the user: while P3P policies are downloaded from well-known location everytime the user's browser connects to a Web site,[35] a user entering a smart environment has no such well-defined interaction that can be used to communicate the data collection practices within this environment. The next section will describe the concept of our *privacy beacons*, which provide such a well-defined disemmination route for our privacy contracts.

[35] Alternatively, both HTTP replies as well as individual Web pages can include a direct link to a P3P policy.

4.4 Privacy Beacons

In our PawS architecture, *privacy beacons* provide the initial link between the data collector, typically the provider of a service in a smart environment, and the data subject, e.g., an individual visitor to that environment, or a user of that particular service.[36]

While Web services can rely on an explicit action on behalf of the user to trigger the download of a privacy policy (e.g., a P3P-enabled browser downloading a P3P policy from the well-known location before actually requesting the user-specified Web page), ubiquitous service do not imply a particular interaction pattern that can be used to disemminate a privacy policy.

A number of so-called *service-discovery* protocols exist, such as Sun's *Jini* [37] or *Universal Plug and Play (UPnP)* [38], that facilitate the discovery of and connection to services within a specific network environment, e.g., finding the closest or most suitable printer in a client's office. For these cases, we can imagine prescriping a multi-step protocol just as in the case of P3P, where the regular HTTP interaction between Web browser and Web server is extended [82].[39] However, not all future ubiquitous services will follow this pattern of a user explicitly searching for a service over a well-defined interface. PawS instead tries to provide a more general announcement mechanism that actively informs the user of the types of services available in a particular place, building, or room.

This section will describe our privacy beacons in more detail, including their counterpart, the *privacy assistant*, which is responsible for picking up the information from the beacons and relaying it to the (user) privacy proxy described in the previous section.

4.4.1 Requirements

As described in section 4.2.4, PawS differentiates between three kinds of services in ubiquitous service environments:

- *Active services*, where the user has an existing contract agreement with a particular service provider.

[36] Privacy beacons were developed as part of the diploma thesis of Marcel Wassmer [346].
[37] See `www.jini.org`
[38] See `www.upnp.org`
[39] We have prototypically implemented such an approach, i.e., embedding policy information into the regular interaction protocol, into an RFID system, as described in section 6.

- *Optional services* that can be activated by the user through entering into a contract agreement with the service provider.
- *Mandatory services,* which cannot be deactivated by the user but are continuously running.

Privacy beacons must be able to support all three types of services, i.e., a user device must be able to remember prior agreements, detect optional services, and be informed about mandatory data collections in order to provide the user with at least a summary of the data that is collected about her. This implies an *always-on* design, where both privacy beacons and their corresponding receivers must constantly be able to send and receive privacy contract announcements. A number of wireless communication technologies could be used for that purpose:

- *Infrared:* The *Infrared Data Association's (IrDA)*[40] "DATA" standard ("IrDA standard" for short) defines a communication protocol over infrared that is widely supported by mobile devices. While the limited communication range[41] can come as an advantage, as it makes limiting contract announcements on a per-room basis easy, it requires a line of sight between privacy beacon and the user's privacy assistant that is used to pick up the beacon signal.[42] Also, while wall-mounted IrDA senders can be quite powerful, resulting in a range of well over 10 meters,[43] handheld devices have typically a limit sending range due to power restrictions, making a second communication channel for replying from the user's handheld device to the user proxy (or directly back to the service proxy) necessary.
- *Bluetooth:* Bluetooth is a low-cost, short-range wireless communication protocol (up to 10 meters) that provides both data and audio links between computers, mobile phones, and other handheld devices.[44] Bluetooth has the advantage that it does not require a line of sight between sender and receiver, but consumers more power than IrDA (at least on the receiving end of the user device). It can also be detected through physical borders, such as walls and doors, making per-room announcements difficult.

[40] See `www.irda.org`
[41] Infrared communication signals typically cannot penetrate walls, doors, or clothing.
[42] In some cases also reflected signals can be picked up, though this is not reliable.
[43] The devices used in the PawS prototype achieve send ranges of up to 25 meters.
[44] See `www.bluetooth.org`

- *Wireless LAN:* While many mobile computers today feature a built-in Wireless LAN (WLAN for short), it is much less prominent in handheld devices such as PDAs or mobile phones, due to its high power consumption. It has a range of more than 20 meters indoors and up to 200 meters outdoors,[45] also without requiring a line of sight.

- *ZigBee:* ZigBee is an emerging standard for low-power, short-range, low-latency communications, based on the IEEE 802.15.4 specification [173]. While at the time of writing no off-the-shelve components were available to incorporate into the PawS prototype, its combination of short range communication and low power makes it an ideal replacement for Bluetooth, which offers higher data rates at an increased power consumption.

While the modular PawS architecture makes it possible to use any of these protocol, the current prototype uses infrared communication for its privacy beacons as it readily fulfills three important requirements:

1. It offers a connectionless broadcast mode of operation, allowing clients to receive privacy contracts without the need for detecting and connecting to a privacy beacon before.

2. It is readily available in common PDAs, thus facilitating the implementation of a privacy assistant complementing our privacy beacons.

3. It provides us with a simple mechanism to limit contract disemmination, alleviating the need for complex positioning mechanisms (see comments in section 7.2).

However, as infrared is not suitable for routing data back from the mobile device, we rely on WLAN for the backchannel from the privacy assistant to either the user proxy or the service proxy.

4.4.2 Communication Protocol

As described in our list of requirements in section 4.1.3, all privacy-relevant interactions in PawS are handled by privacy proxies. Privacy beacons and the corresponding privacy assistants are responsible for setting up such a communication between two proxies. This is called

[45] See `www.wi-fi.org/opensection/range.asp`

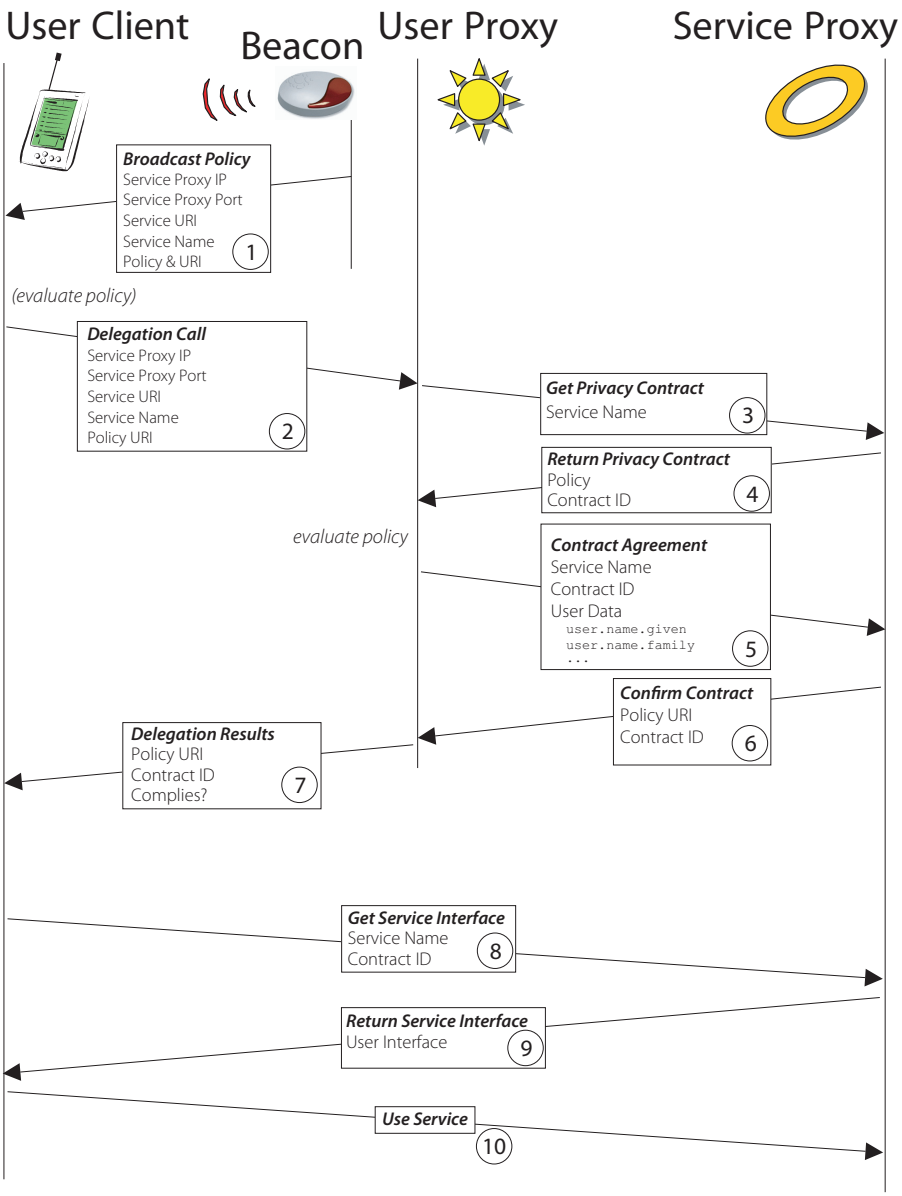


Figure 4.17: *Delegation mode in the privacy beacon protocol.* The typical interaction scenario in PawS assumes that the user's *privacy assistant* picks up a privacy contract from a *privacy beacon* and relays it to the *user proxy*. After successfully agreeing on a privacy contract with the indicated *service proxy*, the user proxy notifies the user's privacy assistant that the service is ready to use.

the *delegation mode*, as privacy assitants pick up the transmission from a privacy beacon and relay it directly to their corresponding user privacy proxy, effectively delegating all further actions to the user proxy. Only if a direct user intervention is necessary (e.g., due to corresponding preferences in the user proxy), as well as after a successful contract acquisition, does the user proxy inform the privacy assistant again (e.g., prompting for a decision, or displaying the results of a service subscription).

Figure 4.17 illustrates the communication flow in delegation mode. The privacy beacon constantly broadcasts the privacy contract for a given service (step 1). The user's privacy assistant picks up the signal,[46] decodes it, and forwards it[47] to the user's proxy (step 2). Note that during decoding, a user's privacy assistant can already filter out known privacy contracts, mandatory services (which do not support contract agreements), or even those incompatible to the user's preferences[48] and immediately end the protocol in order to save energy. Steps 3 through 6 depict the proxy interaction described in the previous section, assuming an acceptable privacy contract given the user's preferences (as stored at the user proxy). Once an agreement has been reached, the user proxy then notifies the privacy assistant whether the service has been accepted or not (step 7). This for example could enable the privacy assistant to directly request the service-interface from the service proxy (steps 8 through 10).

As an alternative communication model, PawS privacy assistants also support the *proxy light mode*, in which they directly provide the services typically offered by the user proxy. This can be helpful in situations when no direct connection to the user's privacy proxy is possible, or if for efficiency reasons a direct interaction via a short-range wireless communication technology is preferred. Under such circumstances, the user's privacy assistant provides in principle the same service as the user's privacy proxy, though with limited functionality. The privacy assistant would typically cache its direct interaction with the service proxy, and later synchronize itself with the user proxy.[49] Advocates of wearable computing might even consider running the entire user proxy directly on the user's mobile device, given sufficient computational capabilities and battery life – the PawS architecture does not require user proxies to run on a specific machine or at a specific location in the network.[50] Figure 4.18 shows the corresponding communication protocol for the proxy light mode.

Instead of contacting the user proxy after receiving the beacon signal,

[46] In the current prototype, this is done using the built-in infrared receiver of a PDA.
[47] Via WLAN in the current system.
[48] More powerful mobile devices might mirror the user's privacy preferences from the user proxy in order to evaluate received privacy contracts on the spot.
[49] Synchronization has not been implemented in the current prototype.
[50] Note, however, that a direct wireless interaction between user proxy and service proxy reveals the user proxy's network identification, e.g., its MAC or Bluetooth address. For increased protection, such protocols should also be anonymized at the network level.

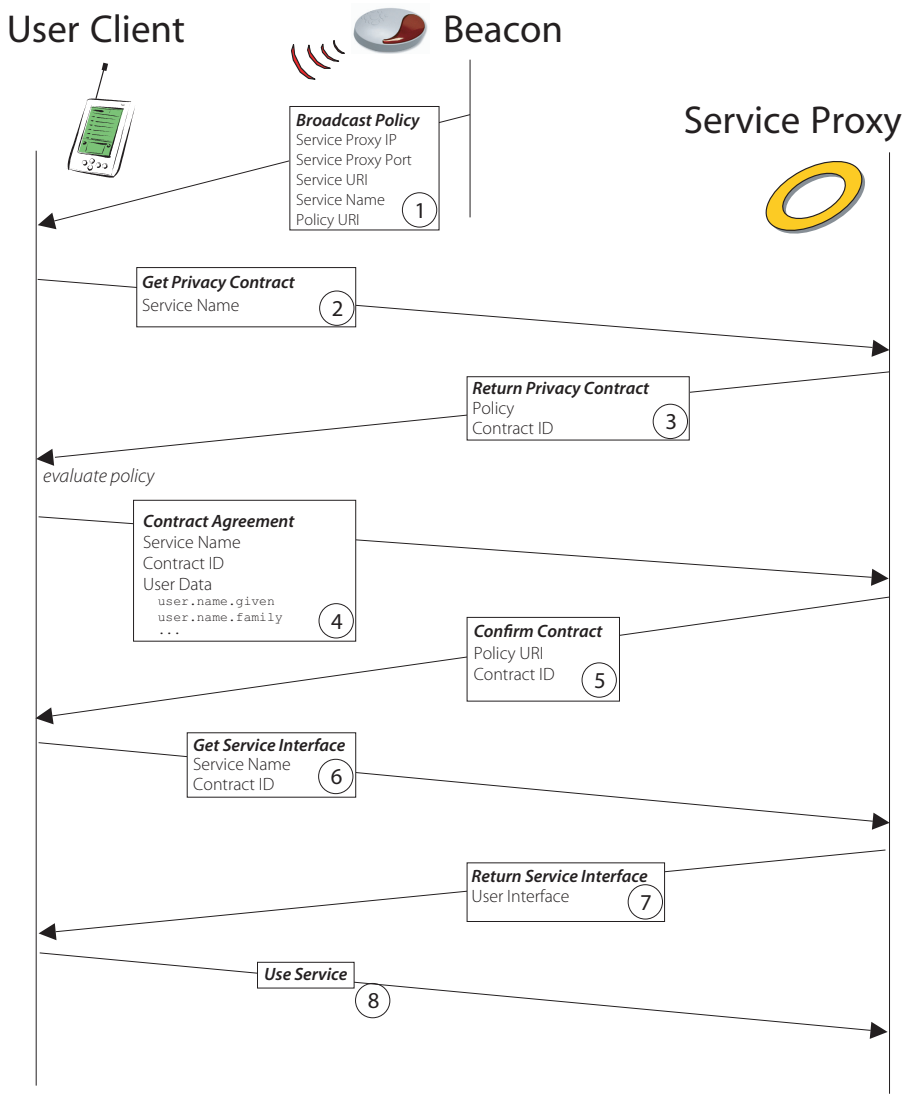


Figure 4.18: *Proxy-light mode in the privacy beacon protocol.* Alternatively, the user's *privacy assistant* can directly interact with the *service proxy*, e.g., if no direct connection to the *user proxy* is available, or if a short-range communication is preferred for power efficiency purposes.

the user's privacy assistant directly contacts the service proxy[51] (step 2), from which it receives the contract for evaluation.

In case the mobile privacy assistant already knows the contract ID that it received from a beacon, i.e., should the corresponding service already be *active*, the communication can be significantly shortened, as the privacy assistant would directly start with steps 6 or 8 in figure 4.18, depending on whether it previously cached any user interface.

[51] Using WLAN in our current prototype.

4.4.3 Signalling Format

A beacon signal contains simply a complete PawS privacy contract, as described in section 4.2 above. It is continuously broadcasted using a probabilistic protocol, i.e., each signal has a certain probability of being signalled during a single signalling cycle. This allows beacons to advertise more than a single policy, while at the same time weighing the relative importance between them. Privacy assistants that enter the vicinity of a beacon will thus pick up more important policies with a higher probability, while less important announcements might take longer to be registered at the client side.

For example, a signal describing the use of a mandatory security camera as well as an optional follow-me phone service might decide to register two separate announcements with the privacy beacon, giving the security camera announcement a significantly higher weight than the optional phone service, thus shortening the time until a client is able to pick up the camera announcement.

This approach specifically takes into account short and intermittent client connectivity times, slow data rates, multiple policy announcements, and unreliable communication protocols.

4.4.4 Implementation

PawS uses infrared beacons developed as part of the *IRREAL*[52] project of the University of Saarbrucken [29], as they combine a large transmission range (up to 20 meters) with a wide transmission angle. They are also relatively simple to setup and were readily available through a personal contact with their developers. Figure 4.20 shows the actual beacon hardware used in the PawS prototype.

The beacons are connected via a serial cable to a computer running a beacon deamon, `ird`, which provides the communication interface to the beacon. Upon registration of a new pricy contract, the service proxy uploads the contract together with its signalling probability (set by the service administrator) to the `ird` deamon. The deamon takes the probability weight of each registered contract and computes its percentage of the total weights, thus balancing the probability that each contract is sent during a send interval.

[52] IRREAL stands for "Infrared REAL" and is an indoor localization system based on infrared senders and receivers. It is part of the REAL project (Resource Adaptive Localization) [341], which also features an outdoor localization component, ARREAL ("Augmented Reality REAL").

```
<POLICIES xmlns="http://www.w3.org/2001/09/P3Pv1">
  <EXPIRY date="Thu, 31 Dec 2002 23:59:59 GMT"/>
  <POLICY discuri="http://www.mywebcam.ch/policy.xml"
    name="WebCamPolicy">
    <ENTITY>
      <DATA-GROUP>
        <DATA ref="#business.name">The WebCam Company</DATA>
        <DATA ref="#business.contact-info.online.email">big@brother.ch</DATA>
        <DATA ref="#business.contact-info.online.uri">http://www.webcam.gov</DATA>
        ...
      </DATA-GROUP>
      <EXTENSION optional="yes">
        <SERVICE name="Web Cam Service"
          xmlns="http://www.w3.org/2004/02/PrivacyContract_beacon"
          type="continuous-collection-service" mode="optional" >
          <DESCRIPTION>We publish the video on the internet</DESCRIPTION>
          <SERVICE-URI-ADDRESS>129.132.178.93</SERVICE-URI-ADDRESS>
          <SERVICE-URI-PORT>6080</SERVICE-URI-PORT>
          <SERVICE-URI-PATH>/interface/servlet/ServiceProxy</SERVICE-URI-PATH>
          <SERVICEMODULE-CLASSNAME>TestModule</SERVICEMODULE-CLASSNAME>
          <info/>
          <consulting/>
        </SERVICE>
      </EXTENSION>
    </ENTITY>

    <ACCESS>
      <all/>
      <EXTENSION>
        <ACCESS-METHODS xmlns="http://www.w3.org/2004/02/PrivacyContract_beacon">
          <UPDATE version="PRO2-1.0" rpc_uri="http://localhost:6080/soap/servlet/rpc"
            service_urn="UserAccess">
            <DATA-GROUP xmlns="http://www.w3.org/2001/09/P3Pv1">
              <DATA ref="#biometry.optical.video" />
            </DATA-GROUP>
          </UPDATE>
          <DELETE version="PRO2-1.0"
            rpc_uri="http://localhost:6080/soap/servlet/rpc"
            service_urn="UserAccess">
            <DATA-GROUP xmlns="http://www.w3.org/2001/09/P3Pv1">
              <DATA ref="#biometry.optical.video" />
            </DATA-GROUP>
          </DELETE>
          <QUERY version="PRO2-1.0"
            rpc_uri="http://localhost:6080/soap/servlet/rpc"
            service_urn="UserAccess">
            <collection/>
          </QUERY>
        </ACCESS-METHODS>
      </EXTENSION>
    </ACCESS>

    <DISPUTES-GROUP>
      <DISPUTES resolution-type="independent"
        service="http://www.resulution.com"
        short-description="ServiceController">
        <LONG-DESCRIPTION>  ...  </LONG-DESCRIPTION>
        <REMEDIES><correct/></REMEDIES>
      </DISPUTES>
    </DISPUTES-GROUP>

    <STATEMENT>
      <CONSEQUENCE>Your video picture will be stored in our database</CONSEQUENCE>
      <PURPOSE><current/></PURPOSE>
      <RECIPIENT><ours/></RECIPIENT>
      <RETENTION><stated-purpose/></RETENTION>
      <DATA-GROUP><DATA ref="#biometry.optical.video"/></DATA-GROUP>
    </STATEMENT>

    <EXTENSION optional="yes">
      <WITHOUT-CONSENT xmlns="http://www.w3.org/2004/02/PrivacyContract_beacon">
        <change />
      </WITHOUT-CONSENT>
    </EXTENSION>
  </POLICY>
</POLICIES>
```

Figure 4.19: *Example of a PawS beacon message,* which in effect is a regular privacy contract. The important information when forwarding such a message to a user proxy is the service data contained in the `SERVICE` element.

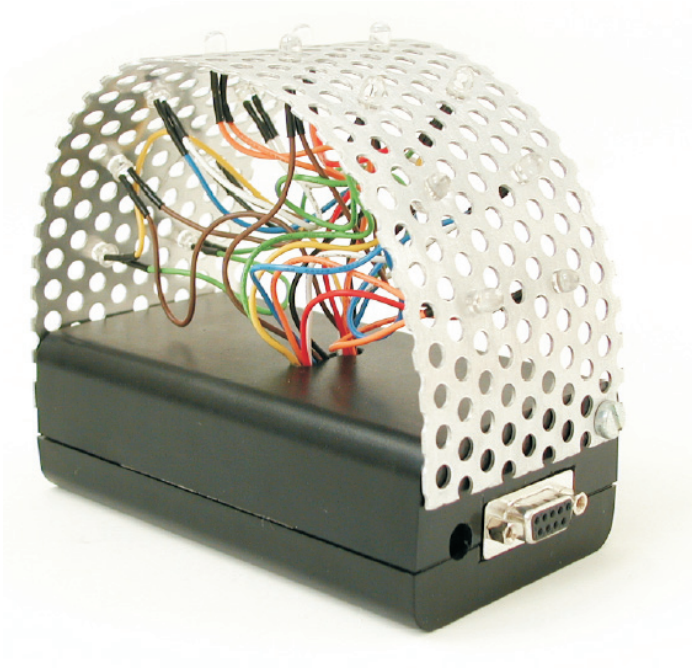
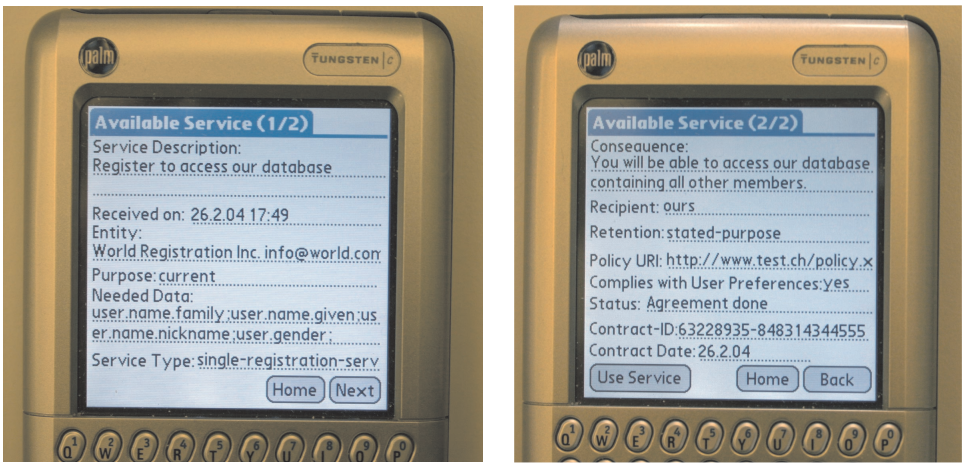


Figure 4.20: *Privacy beacon hardware,* taken from the *IRREAL* project developed at the University of Saarbrucken [29]. It is an infrared sender with 10 LEDs that is connected via an RS232-interface to a PC. The signal range is up to 20 meters, with a customizable signal angle.

Figure 4.21: *Privacy assistant main interface,* as implemented on a Palm PDA. It shows the services for which a privacy contract has been received from a beacon, as well as a list of currently active services.



(a) Service Details Page 1 (b) Service Details Page 2

Figure 4.22: *Privacy assistant service details* are displayed on two separate pages. In order to enter into an agreement, the user presses the "use service" button (b). If the service would already be active, a "discontinue" button might appear instead, should the service support this.

On the user side, we have implemented a simple user interface to both visualize beacon activity, and to interact with the overall system. Figure 4.21 shows the main overview page of the application. At the top of the screen, a list of privacy contracts that have been received from privacy beacons is displayed. Below, the application shows the currently *active* services, which might be either those for which an explicit agreement has been reached, or mandatory services such as a surveillance camera, for which the system simply tracks data collection. Due to the lack of multitasking support in the Palm OS, scanning must either be done manually through a menu item, or by selecting the "Automatic Scan" checkbox, which prompts the application to continuously look for beacon messages on its infrared port, at the expense of menu reactivity and battery power.[53]

Selecting an entry from either list and clicking on the "details" button, the user is taken to a simple enumeration of the privacy contract details, as shown in figure 4.22. Should the user decide to use one of the offered services, she can click on the "Use" button (see figure 4.22.b) in order to enter into an agreement with the service, based on the offered contract. Similary, currently running services can optionally be ended from within this application, and user data updated.

The implementation on the Palm PDA uses the *Cytheric XML Li-*

[53]For interaction with the application, it is therefor necessary to deselect the automatic scan feature.

brary[54] for parsing the beacon signals (i.e., the privacy contracts) and the standard Palm OS functions for contacting the user proxy (in *delegation mode*) or the service proxy (in *proxy light mode*) via the built-in Wireless LAN. SOAP messages are created and parsed manually using the Cytheric library. No automated decision making has been implemented in the privacy assistant prototype.

4.4.5 Summary

Privacy beacons form a prototypical link between a service proxy's policy announcements and an individual user entering the service's vicinity. Beacon signals are picked up by a mobile device, the user's privacy assistant, which either processes this information itself (resulting in a direct exchange with the service proxy, or simply a logfile entry if no communication is necessary), or forwards it on to the user proxy for processing.

The current prototype uses infrared beacons and a regular Palm PDA. While this has the advantage of allowing ranged (e.g., room-sized) policy announcements, it requires a line-of-sight between the PDA and the beacon, prompting the user to actively 'sweep' the privacy assistant to pick up the signal.

4.5 Privacy Database

While the initial privacy contracts and all configuration data is individually managed by the proxies using the integrated XML database, a separate component is responsible for storing the collected personal information on the service side: the *privacy-aware database*, PawDB for short.[55]

The basic idea of a privacy-aware database is that all data accesses are done in accordance with the privacy policy that governed the initial data exchange. This means that upon storage of new data, not only the data itself but also the privacy policy that describe its allowed usage, disemmination, and retention, must be stored along with the data. Similarly, upon receiving a query, a privacy-aware database requires a policy declaration that describes who the recipient is and under what conditions this queried data is to be used. The query then only returns

[54] See `cytheric.net/palm-xml`

[55] The PawDB prototype was developed as part of the diploma thesis of Paul Miotti [244].

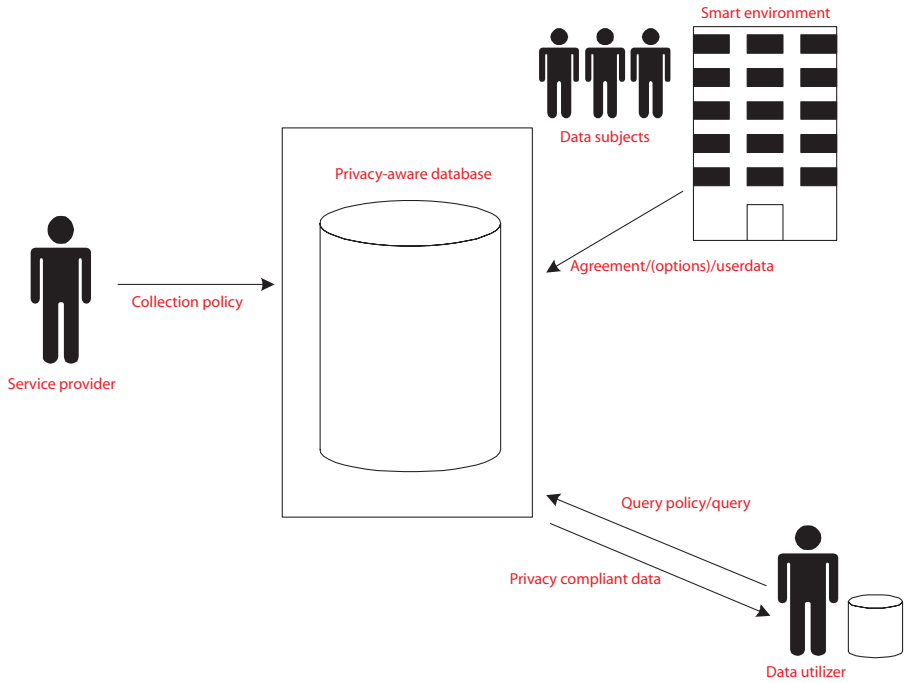


Figure 4.23: *PawDB overview.* All data access within PawDB requires a *query policy* declaration, which needs to be compatible to the data elements original *collection policy* in order to allow the data element to be returned as part of a query to a party wanting to utilize the information.

data elements whose collection policy matches the requester's policy declaration. Figure 4.23 gives an overview of this.

As a full-fledged database model and implementation is beyond the scope of this work. Instead we have opted for a prototypical proof-of-concept system, for which we will illustrate the overall data model, discuss the requirements for policy management, and explain how retention enforcement works. Then we will briefly outline the implementation as part of PawS, before closing this section with a summary.

4.5.1 Data Model

PawDB differentiates between two types of policies: *collection policies*, i.e., the privacy policies that were the basis for the initial data collection, and *query policies*, i.e., the declarations by database users at query time in which they specify the particular purpose for the query, etc. These two policies must be compatible in order for a query to return a particular data element in its results.

Compatibility is defined in terms of data usage, i.e., the query policy's data usage – its purpose, recipient, and retention specification – needs to be subsumed by the collection policy's data usage in order to allow

for the inclusion of the data in a query response. These three policy attributes thus translate into the following three requirements:

1. *Purpose match:* A query must only return data elements if the *query policy*'s purposes are a subset of the purposes declared in the data elements' *collection policy*.

2. *Recipient translation:* A query must only return data elements if the relation between the *query policy*'s `ENTITY` element and the *collection policy*'s `ENTITY` element matches the collection policy's recipient declaration, *and* if the query policy's `RECIPIENT` element is subsumed by the policy's `RECIPIENT` element.

3. *Retention limitation:* A query must only return data elements whose retention period (as specified in the *collection policy*) is equal or greater than the retention period specified in the *query policy*.

While both the purpose match and the retention limitation are straightforward comparisons, the recipient match requires a lookup table in the database that enumerates all possible entities in the system and their relationship (in terms of P3P vocabulary) to the data collector. Consequently, data access from entities not listed in the lookup table must be ignored.[56]

The comparison is further complicated by the ability to declare both optional purposes and recipients in a collection policy.[57] For efficiency reasons, the individual user choices for a collection policy are not translated into a separate policy but instead collected in a per-contract options table. Consequently, comparisons between query policies and collection policies need to take into account the actual user choices regarding any optional purposes or recipients as per the corresponding privacy contract.

4.5.2 Policy Management

Data in PawS typically comes in the form of XML documents, such as the privacy contracts or the submitted user data. In order to store

[56] A certificate-based authentication scheme should be used to verify the validity of the entitiy declaration in the query policy.

[57] As no user interaction is possible at query time, *query policies* are not allowed to declare optional purposes or recipients. All present declarations are assumed to be mandatory in a query.

and later query such data, a number of native XML database systems have been developed, such as the Xindices system used in our privacy proxies. However, the drawback of such systems is their limited performance and query capabilities when compared to existing relational database systems. A number of techniques have been proposed to store and query XML documents in relational databases [121, 318], which typically involves three steps:

1. *Relational schema generation,* in which relational tables are created that ultimately hold the information present in the XML documents.

2. *XML document shredding,* which involves parsing the documents that should be stored and storing each individual elements into rows in the previously generated tables.

3. *XML query processing,* in which XML queries are translated into SQL queries that operate on the relational table data.

Using a relational database system in PawS thus involves a two-step setup phase, in which the database tables are first initialized with the XML schema definition of the privacy contracts and user data sets,[58] before each individual collection policy that is offered by the service proxy must be registered with, i.e., shredded into, the database. Similarly, once personal data is submitted to the database, this information is shredded into its corresponding relational table row. The following four types of XML documents must thus be translated into relational formats:[59]

- *Collection policies,* i.e., privacy contracts from the service proxy. These form the templates for the individual agreements under which each data element is stored. Each policy is identified by a unique `policy_id`.

- *User contracts,* i.e., accepted privacy contracts that can potentially be configured along a variety of optional choices, such as optional recipients or purposes. Each is referred to by an individual `agreement_id` and references a template `policy_id` of the original collection policy.

[58] XML schemas are used to define XML vocabularies that can be used to write shared documents [113].

[59] The XML document shredding and query processing would typically be handled transparently by a corresponding XML-translation layer in the RDBMS.

- *User data sets,* as submitted from the user proxy or a privacy assistant to the service proxy. Each data set is stored under its governing `agreement_id`.

- *Query policies* can optionally be registered for performance improvement, allowing certain user groups, e.g., marketing or controlling, to choose from a selection of readymade queries. Queries are registered under a `query_id` and can link to a complete SQL-query that implements the data model described above.

Query policies are similar to regular PawS privacy contracts, i.e., they are enclosed in a `POLICY` keyword and feature an `ENTITY` element, one or more `STATEMENT` elements, etc., with the following exceptions:

- Query policies do not contain the `ACCESS` or `SERVICE` extensions described in section 4.3.1 above.

- Query policies do not contain a `DISPUTES-GROUP` element.

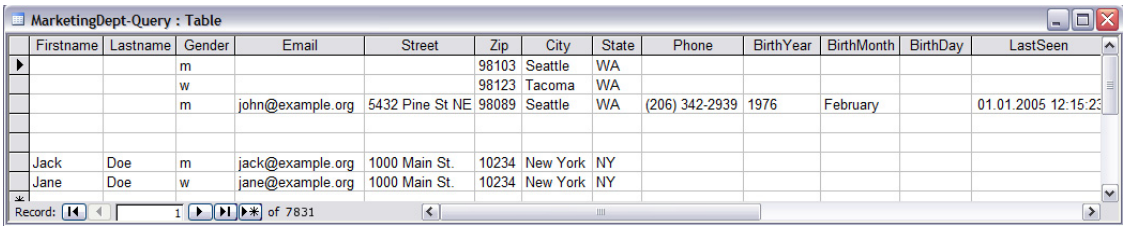
The elements that are to be queried are implicit in the query policy declaration, corresponding to an SQL `SELECT *` statement. For more complex queries, an automated mechanism could dynamically create individual query policies from a template query policy (featuring only the fixed data such as the entity declaration, purposes, and recipients) and a complex SQL query.

After issuing the query, its policy is either taken from the list of preregistered, shredded query policies, or dynamically translated into a meta query that combines the "real" query with the policy matching described in the data model above. A possible reply to such a query can be seen in figure 4.24.

This functionality is realized through a dedicated API layer that can directly be accessed by the service proxy, or through specialized forms for direct terminal interaction, e.g., in an individual department where queries are entered. The following core interfaces need thus be provided:

- *Collection policy upload.* Data collectors (through their service proxies) must register privacy contract templates with the service. Instances of these templates will later be referenced from the submitted user data.

- *User data storage.* Users store personal information under a specific privacy contract instance (using the service proxy as a relay).



MarketingDept-Query : Table

Firstname	Lastname	Gender	Email	Street	Zip	City	State	Phone	BirthYear	BirthMonth	BirthDay	LastSeen
		m			98103	Seattle	WA					
		w			98123	Tacoma	WA					
		m	john@example.org	5432 Pine St NE	98089	Seattle	WA	(206) 342-2939	1976	February		01.01.2005 12:15:23
Jack	Doe	m	jack@example.org	1000 Main St.	10234	New York	NY					
Jane	Doe	w	jane@example.org	1000 Main St.	10234	New York	NY					

Record: 1 of 7831

Figure 4.24: *PawDB example query results,* in which a marketing department query returns personal information only from those data subjects who explicitly permitted a purpose of `<marketing/>` in their privacy contract. This can also be done on a per-element level, resulting in "spotty" replies that feature regions of unavailable data. In the example above, empty cells mean that either the data elements are not permitted to be used for marketing purposes, or they are not available.

User data must thus always be submitted together with a reference to a privacy contract template and the user's specific choices for optional values.

- *Query policy upload.* In order to retrieve data from the database, individual query policies need to be registered under which queries can be performed later.[60]

- *Data usage through queries.* A query interface handles SQL or XML queries over user data under a specific query policy (which has been previously uploaded).

Figure 4.25 summarizes these access requirement in four corresponding interface specifications. Instead of accessing the underlying database directly, all data access is routed through a specialized privacy-API that limits information disclosure to policy compliant uses and recipients.

4.5.3 Retention Enforcement

Besides enforcing matching purposes and recipients, PawDB also takes care of deleting (or anonymizing) personal information with an expiring retetion period. Purging can be done on an ad-hoc basis, i.e., a special retention module explicitly filters out expired information whenever data access is taking place, earmarking it for immediate deletion through a separate process.

An alternative approach, which also does not impact query performance, is to periodically check stored data for upcoming expiration. The maintenance cycle needs to be equal or shorter than the retention

[60] This is mainly an efficiency optimization.

```
PolicyID register_collection_policy (String xml_policy)
ContractID retrieve_privacy_contract (PolicyID collection_policy)

PolicyID register_query_policy (String xml_query_policy)
```

(a) Policy management methods

```
int store_user_data (ContractID contract_ref, String xml_userdata)
```

(b) User data storage methods

```
QueryResponse xml_query (PolicyID query_policy, String xml_query)
QueryResponse sql_query (PolicyID query_policy, String sql_query)
```

(c) Data query methods

Figure 4.25: *PawDB access methods.* Policy management methods allow the storage and retrieval of both *collection policies* and *query policies*. Users store personal data under their individual *contract agreement ID*. In order to query information, the governing query policy must be referenced.

resolution in order to provide timely deletion services, e.g., once a day (preferably in the early hours of the day) for daily expiration times. Such a process could also explicitly schedule deletions upon encountering a soon-to-expire retention date. This might also be preferable if data is not to be deleted, but rendered anonymous, which might entail more computationally intensive (and thus slower) processes, for which some sort of scheduling might be necessary.

The standard P3P retention periods use generic identifiers such as `stated-purpose` or `business-practices`, prompting the data subject to follow a link to a web page with detailed information about the data destruction timetable [79]. PawDB must use a translation table that maps such identifiers to a time and date relative to an absolute timestamp associated with a particular data item, such as its collection time or a billing date. Using the P3P extension mechanism, we can also directly embed such times into our privacy contracts, using either a relative date (e.g., "90 days") or an absolute date and time (e.g., 3 weeks after a prize drawing in a lottery). Table 4.26 gives examples of each of the three options, with our extension using the HTTP/1.1 date conventions [116] for absolute and relative time.

```
<RETENTION>
  <legal-requirement/>
</RETENTION>
```

(a) Identifier-based retention period

```
<RETENTION>
  <business-practice/>
  <EXTENSION optional="yes">
    <delta-seconds>7776000</delta-seconds>
  </EXTENSION>
</RETENTION>
```

(b) Relative retention time (extension)

```
<RETENTION>
  <stated-purpose/>
  <EXTENSION optional="yes">
    <full-date>Sun, 06 Nov 1994 08:49:37 GMT</full-date>
  </EXTENSION>
</RETENTION>
```

(c) Absolute retention time (extension)

Figure 4.26: *PawDB retention times.* Retention times in a privacy contract can use an identifier as defined in the P3P specification [79] or a relative or absolute data, given as an extension to the required P3P identifier (using the format defined in [116]).

4.5.4 Implementation

A prototype PawDB has been implemented in Java 1.2 on top of an *Oracle8i* database[61] running on a Linux machine. It uses the Oracle *Java Database Connectivity* (JDBC) drivers[62] to implement an intermediate API that provides the methods described in the previous sections. The API layer is comprised of some 50 classes and some 9000 lines of code. Storing ("shredding") XML data in the relational Oracle database is done with the help of XML-DBMS.[63] Figure 4.27 shows an excerpt of tables generated for storing PawS privacy contracts. Similarly, figure 4.28 shows the tables created for storing XML user data.

4.6 Discussion

The work presented in this chapter is only an initial prototype for how a technical privacy system that takes into account both social and legal mechanisms might look like. As such, the presented work does leave a number of issues unaddressed and suggests several avenues for future research.

4.6.1 Limitations

PawS main limitation lie in the general field of usability: as only few developers have been testing the overall information flow between privacy proxies, beacons, and assistants, a real-world deployment of PawS would require a number of improvement.

P3P Issues

The use of P3P as a privacy tool is not undisputed. Critiques of P3P such as Clarke [63] and Catlett [58] see the use of such protocols as a way to delay attempts at properly regulating privacy in the US. They also point out that P3P *facilitates* data exchange, rather than provide privacy through restricting it, thus commodifying individual privacy and encouraging increased "selling" of personal information.[64]

[61] See `www.oracle.com`

[62] See `java.sun.com/products/jdbc/` and `www.oracle.com/technology/tech/java/sqlj_jdbc/htdocs/jdbc_faq.htm`

[63] XML-DBMS is a middleware for transferring data between XML documents and relational databases, mapping XML data according to an XML document's DTD. See `www.rpbourret.com/xmldbms/`

[64] See our discussion of privacy as property in section 3.2.1.

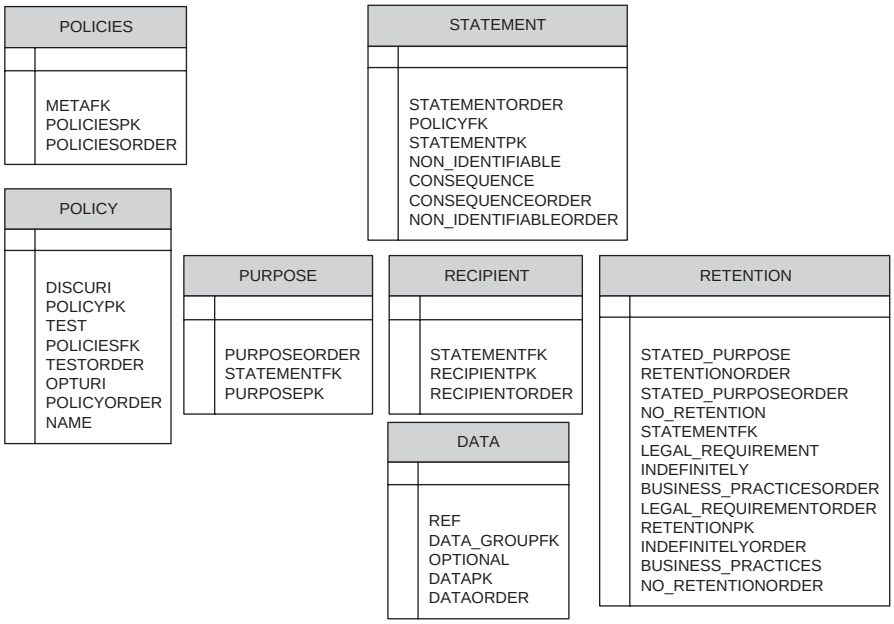


Figure 4.27: *Tables for storing privacy contracts in PawDB (excerpt).* Each registered privacy contract is distributed across several tables, each corresponding to an XML element in the policy. Table creation and distributed data storage is done using XML-DBMS, an open-source middleware for transferring data between XML documents and relational databases.

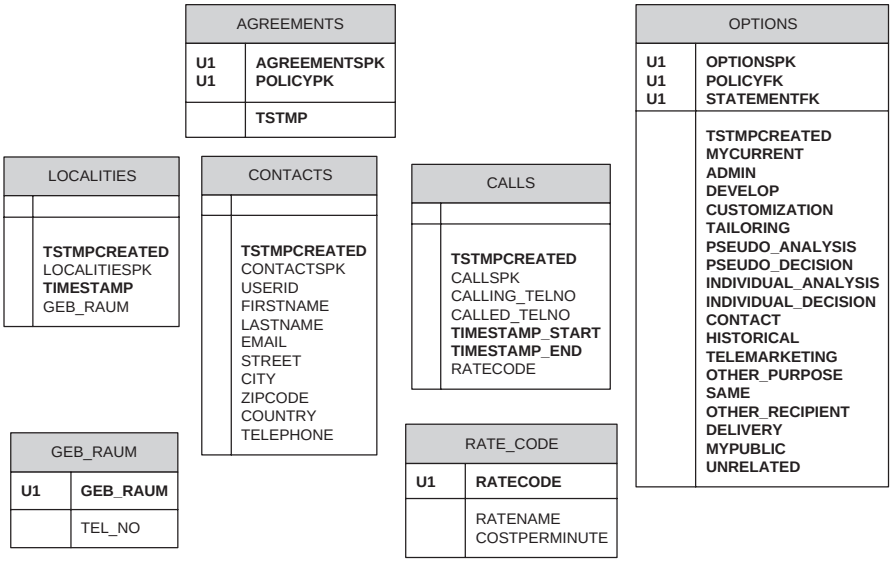


Figure 4.28: *Tables for storing user data in PawDB (excerpt).* Similar to stored privacy contracts, the personal information of service users is stored in distributed tables representing XML documents of the user's personal information.

In particular, the lack of specific access and retention information in the original specification is seen as key elements for privacy protection that are absent from P3P.

While our work explicitly points out the need for social and legal frameworks in which to embed a technical tool such as PawS, the use of such systems in contexts that lack such supporting mechanisms might be problematic. Also, PawS explicitly extends P3P to include detailed access and retention information, though the acceptance of such declarations in the marketplace might again be contingent on clear legal requirements.

Hochheiser [166] further points out that the P3P vocabular might appear simple and clear to users, yet that the terms, as used by services, often are restricted or have meanings that are not obvious. Ackerman [4] remarks that this is an inherent tradeoff present in many user-centric technologies, as it involves a conflict between a vocabulary that is brief and understandable vs. a vocabulary that is complex but completely explanatory.

User Preference Specification

The PawS prototype does not support taking automated decisions based on a user's privacy preferences, nor does it allow users to specify such preferences. Each detected service must be manually subscribed to, as show in figure 4.21 on page 166 above. An obvious choice would be to implement an APPEL rule evaluator [80], similar to the AT&T Privacy Bird implementation described in section 5.1.2.[65] However, several alternative, simpler approaches to a privacy preference interface exist, see section 5.1.2 for an overview.

Multi-User Preferences Reconciliation

All scenarios in PawS assume a single user utilizing one or more ubiquitous services. However, in real world settings, several users will most likely be co-located and share a common set of services. This makes a *reconciliation* of differing preferences necessary if services involve sensors that operate, e.g., on a per room basis. A simple example would be a lecture hall that provides audio and video recordings of lectures. While some students might want to subscribe to a service offering a

[65] A more comprehensive APPEL rule editor has been developed by the EU's Joint Research Center, see `p3p.jrc.it/downloadP3P.php`

complete recording of the lesson, others might object to being recorded on video or having the questions or comments saved.

Existing classroom systems such as the Classroom 2000 project at the Georgia Institute of Technology[66] use a "social" solution: video recordings are taken from behind the students, so only the face of the lecturer is recorded [1]. Other possibilities would be a selective blanking out of individual faces in the video stream [315] or a clearinghous approach as suggested by Brassil [47], in which individuals can register time and space coordinates with a central services that allows for the removal of not-to-be-released footage from submitted (time and space tagged) video. While an automated solution looks appealing, the simplest and maybe most effective approach would be to rely on social norms to prompt participants in a meeting or lecture to communally arrive at a decision, similar to, say, deciding whether to open a window, turn on the air condition, or lowering the light levels.[67]

Social Interaction

PawS only focuses on the interaction of a user with a (presumably) commercial or institutional service (e.g., a building's security system). However, ubiquitous computing applications often work in social settings, e.g., to bridge distant familiy members, or to provide awareness to a group of friends. Wearable computers that allow their wearers to keep a multimedia diary by continously recording a video and audio stream from the user's point of view [289] are another example of such interactions.

While this is an important aspect of a ubiquitous computing future, such interaction might require a very different approach in order to preserve individual privacy. Instead of focusing on technical and legal solutions (e.g., machine-readable privacy policies), this might be much more of a user interface issue: How easy is it for users to control their current visibility? What kind of interactions will be deemed socially acceptable in the future? And how simple is it to turn services selectively on and off?

[66] The project has since been renamed to "eClass," see `www.cc.gatech.edu/fce/eclass/`

[67] A different problem is peer-capturing, i.e., instead of a central room infrastructure, each participant might decide to turn on his or her personal recording device. See the subsection on social interaction below.

Negotiation

While by some seen as a limitation, negotiation is deliberately missing from our architecture. Our simple policy announcement and selection mechanism provides users with an up-front view on *all* of the available options, instead of forcing them to haggle with an automated process in order to get the "best" deal (and never knowing whether they really got it). While some users might find it desirable to negotiate for example the amount of rebate they receive when giving out their personal data, we speculate that for most businesses the costs for creating and maintaining such complex negotiation engines will be greater than their benefits.

4.6.2 Strengths

Despite its limitations, the approach chosen in our prototype system has also a number of advantages over comparable systems as discussed in the next chapter.

Minimal Usage Effort

PawS is designed to require the user's assistance as seldom as possible. While its current form does not support preferences, the overall architecture envisions a similar usage scenario as the original P3P specification [81]: Starting from a set of predefined rules, e.g., as provided by some governmental or international agency such as the EU, a user only actively changes her personal ruleset if she wants to use a particular service. All other times, she could safely ignore the information PawS keeps track of for her. It is like the official "terms of business" printed in a mail order catalogue: even though few customers look at them, companies should not be released from their obligation to post them. That is because it is such public display that ultimately brings about *accountability*. Data protection officers or consumer watchdog organizations could for example take a random sample every so often by walking around and comparing PawS announcements with legal requirements or public statements of a service provider. Simply the threat of being held accountable for making false statements is a force often much more powerful than technical locks that can eventually be circumvented.

Failsafe Operation

The service model in PawS assumes an explicit consent that is required for all data collections, unless a mandatory service is allowed to do so by law. Thus, even if a personal privacy assistant fails, users will receive a level of privacy protection comparable to today's level. While a defective device will fail to record all mandatory data collections taking place, the overall loss should not affect an individual's privacy.

Compatibility with P3P

As PawS builds upon and extends the P3P standard, it can directly make use of related tools and libraries, such as JRC's APPEL ruleset editor.[68] It also benefits from the substantial legal and social expertise that has been put into the development of this standard.

4.7 Summary

In this chapter we have presented the prototypical architecture of PawS, a privacy-awareness system suitable for supporting the individual in a world full of ubiquitous sensors and services. Using an existing machine-readable format for privacy policies on the Web (P3P), extending it with detailed access and location descriptions, and disemminating it wirelessly using privacy beacons, we can provide a mechanism for giving proper *notice* to the data subject. We have implemented privacy proxies as a set of Web services to support *choice and consent*, and extended a standard database system with privacy-metadata mechanism (PawDB) in order to allow for *access and recourse*.

Privacy proxies form the core elements of our architecture: whenever the user wants to utilize a certain service that requires personal information to be submitted in order to function (e.g., a tracking services that allows telephone calls to be routed to the telephone at the user's current location), the *user proxy* contacts the *service proxy* at a URI published either as part of a service protocol (e.g., Jini, or as part of an RFID reader-to-tag protocol) or a continuously running *privacy beacon*. The service proxy replies with one or more *privacy contracts*, indicating various levels of service offered and the data needed in each case. Depending on the user's preferences, the user proxy then selects one such policy and replies with the relevant data, using XML messages

[68] See `p3p.jrc.it/downloadP3P.php`

embedded in SOAP calls. Upon successful completion of the interaction, the service proxy replies with an *agreement ID* that is kept by the user proxy for reference.

Depending on each individual agreement, clients can at any time after the data exchange use the agreement ID to inspect the personal information stored with the service proxy, or request updates or deletion of their personal data through optional *PRO2 access methods*. Messages can be digitally signed using *SOAP signatures* and are sent using HTTP over SSL to prevent eavesdropping. Authentication is simply done using the agreement ID created from the actual data exchange and returned to the client as part of the exchange protocol.

A privacy-aware database, PawDB, stores data collected by the service proxies under the individual contract agreement IDs, each linking to the original *privacy contract* that was offered by the service proxy. In order to query any of the stored data, a corresponding *query policy* must be submitted together with the query, which describes in detail the entity requesting this data, the purpose of the query, and how long this information is stored in turn. The PawDB system then compares each query and its query policy to the collection policy of each individual element and transparently withholds a particular piece of information in case of a mismatch. Furthermore, a daemon process takes care of the guaranteed storage periods set out in the original data collection policies.

5 Related Work

There's more than one way to do it.
Larry Wall[1]

In this chapter we want to survey and discuss alternative approaches to privacy in ubiquitous computing – both from within the field, as well as in related areas. Since our initial article on privacy at Ubicomp 2001 [205], privacy issues have gradually become a staple in ubiquitous computing conferences, leading to a number of alternative solutions than what we have presented with PawS. The following sections try to assess these proposals and contrast them with our own work. We will first present a number of general privacy tools for ubiquitous computing, including infrastructures, identity management, and data management architectures. A separate section will look at the fields of *location privacy* and *RFID privacy*, briefly summarizing the issues and the currently proposed solutions.

5.1 General Tools

There is by now a wide variety of technical privacy tools and systems related to ubiquitous computing proposed in the literature. The following sections try to briefly describe work that is close enough to PawS in scope and implementation, though the exact line is difficult to draw. Also, some of the work described below falls into multiple categories, e.g., an infrastructure for location privacy that uses identity management (e.g., [249]), in which case the most relevant aspect has been used for classification into the enumeration below.

5.1.1 Privacy Infrastructures

PawS is not the only attempt at making smart environments more privacy friendly. The following lists a number of alternative attempts,

[1] See `en.wikiquote.org/wiki/Larry_Wall`

as well as some early pioneering work, and contrasts it with our own approach.

The Confab Toolkit

A recent privacy-aware ubiquitous computing infrastructure is the *Confab Toolkit* by Hong and Landay [167]. Originally starting out as a programming framework for context-aware applications, it has since added explicit privacy mechanisms to its data management tools.

Data in Confab is managed in *InfoSpaces*, network-addressable logical storage units that store context information about a single entity, i.e., a person, a location, a device, or a service. *In-* and *out*-filters manage data flows between different infospaces, with *in*-filters only allowing the storage of data from trusted sensors or entities, and *out*-filters enforcing access policies and adding *privacy tags* to all outgoing data.

Privacy tags are similar to privacy contracts in PawS with respect to the idea of using meta data to enforce privacy compliant usage and retention. However, they differ significantly in conception: While privacy contracts are a declaration by the data collector that the data subject basically either rejects or accepts (potentially with a range of options), privacy tags in Confab unilaterally declare what the data subject wants the data collector to do with the data, independently of the data collectors plans.

Privacy tags are also more custom tailored to the exchange of dynamic context data than the P3P-based privacy contracts used in PawS, featuring elements that declare how many "sightings" the other party may amass of a particular attribute (e.g., only retain the last five locations a person was in) and a "garbage collect" declaration that can contain data-deletion triggers, such as when leaving a particular area. In order to provide plausible deniability, information that is deemed too sensitive to be released will be marked by an *out*-filter simply as "unknown," making it indistinguishable from technical failures or lack of connectivity.

The most important difference to PawS lies in the explicit focus on self-captured data in Confab: while PawS addresses smart environments that can communicate and enforce data collection practices for various optional and mandatory data collections, Confab provides a framework for disemminating *locally* gathered context information instead. As such, a combination of the two frameworks for providing complete coverage in smart environments might be desirable.

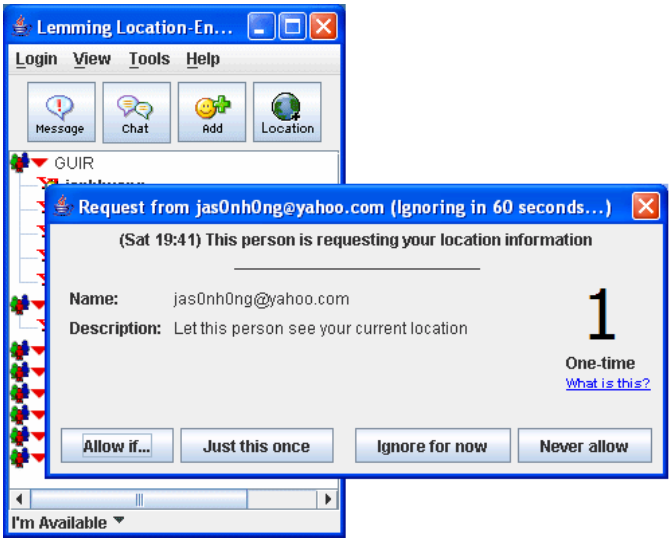


Figure 5.1: *Confab sample application.* "Lemming" is a location-enhanced messenger that is built using the Confab toolkit. Its user interface uses a simple but easily understandable rule-on-demand mechanism, where requests for a location update prompt the user to decide at the time of data collection, rather than ahead of time [167].

Figure 5.1 shows the "Lemming" sample application – a location-enhanced messenger – that has been implemented using the Confab toolkit. While typically running on a laptop rather than a PDA, the interface nevertheless uses similar concepts as our prototypical privacy assistant. A large "1" indicates a one-time disclosure rathen that a continuous query (termed *discrete* and *continuous* collection services in PawS, see section 4.2.4). Users can accept a request for disclosure once, reject it once or forever, or specify a conditional accept rule. In contrast to the examples used in PawS, Lemming does not use data collection policies, i.e., requests for location do not include any information other than the email address of the requestor and a short explanatory description.

The Privacy System by Myles et al.

Myles et al. [249] propose a system very similar to PawS, but with an explicit focus on location privacy. Their core component is a *Location Server* that answers application's requests for a user's location. Users register their privacy preferences at each Location Server using *Validators*,[2] which are consulted by the Location Server before it release any user location. Similar to PawS, Myles at al. extend the P3P schema in order to better describe ubiquitous data collections. The most notable

[2]One for each of their identities.

extension is the `request-initiation` tag, which indicates whether the data is requested due to an explicit trigger by the user, or asked for unsolicited. PawS differentiates between unsolicited and explicitly triggered data collection through its contract references, which allow a privacy assistant to recognize user initiated data collections.

Validators in their system support user preferences over data elements, time, location, and quality, as well as delegating the decision to third-party validator services. A user is assumed to trust the Location Service, her validators, and the positioning infrastructure that feeds the user's location into the Location Service. Anonymity is obtained through providing users with multiple, temporary identities.

Apart from its focus on location services, the system by Myles et al. is otherwise very similar to PawS. Instead of individual user proxies, their approach uses a centralized Location Server component that collects all available user data (and in particular the user's location) and delegates decisions regarding its release to one or more validation components that users must register with the service. Though the authors explicitly see this as a fundamental difference to PawS, i.e., providing privacy checks "at the moment of information release" rather than "at the moment of data capture," they simply move the trust boundary further into the infrastructure. Whereas PawS supports both trusted and third-party positioning systems (i.e., `submitted` and `current` perception data), Myles et al. require the positioning system to be under user control.

Privacy in the Aura Project

CMU's Aura project[3] aims at providing a personal information environment that specifically takes the user's limited attention capacity into account. It also addresses how information collected by the various sensors deployed in the system can be processed and disclosed in a privacy-friendly manner. However, privacy in the Aura project employs largely traditional access control mechanism based on user identity and query context (e.g., time of day) [161]. Also, while PawS assumes a single service provider operating a pervasivce computing environment,[4] access rules in Aura can potentially be enforced at every step in the detection chain.

[3] See `www-2.cs.cmu.edu/~aura/`

[4] Or, alternatively, a number of subcontracted service providers operating individual services in the environment, but contractually bound to the building owner.

Privacy in an Ambient World (PAW)

The *Privacy in an Ambient World* (PAW) project[5] builds upon the results obtained in the PISA project[6] and makes extensive use of digital rights licensing schemes to limit the disemmination of personal information, combined with tools for secure mobile code execution (as the framework relies on mobile agents). The project is still in its early stages, and thus only a set of requirements exist [56].

5.1.2 User Interfaces

PawS does not address the actual user interface for a privacy assistant, even though it remains a very important component in any comprehensive privacy solution. The following lists work in the area of user interface design that addresses both privacy feedback (i.e., communicating to the user her current privacy level) and privacy management (i.e., allowing the user to control her privacy levels).

Privacy in the RAVE System

One of the earliest references to a privacy user interface for a ubiquitous computing system comes from Bellotti and Sellen [33], who tried to provide privacy feedback and control to users in an audio-video presence and collaboration environment called RAVE. Deployed at Xerox's EuroPARC, the RAVE environment consisted of cameras, monitors, microphones, and speakers that were deployed in all offices, and which allowed staff to *glance* at other offices (i.e., get a few seconds of video-only transmission), make *v-phone calls* using both audio and video, or install a longer lasting *office-share* (i.e., a semi-permanent v-phone call). From their experiences with setting up and using such a system, Bellotti and Sellen identified four main problems for users of their system:

- *Capture:* What information is being picked up (e.g., audio or video feeds, still pictures, or work activity)?
- *Construction:* What happens (technically) to this information (e.g., where is it stored or how is it transmitted)?

[5] See `www.cs.ru.nl/paw/`

[6] The *Privacy Incorporate Software Agent* (PISA) was a EU-funded project that used agent technology to provide privacy for e-commerce applications [337]

- *Accessibility:* Who has access to this data (e.g., is it made public, or only available to a particular group)?
- *Purpose:* How will this information be used (e.g., what is the intention of the data collector)?

Based on these user concerns, the authors tried to find both the apropriate feedback and the apropriate control mechanism for each of these problems. Each of these mechanisms was evaluated along eleven criteria, again based on the authors' experience with the design and usage of ubiquitous computing systems:

- *Trustworthiness:* is it a reliable mechanism?
- *Appropriate timing:* does feedback come at a time when control is required and effective?
- *Perceptability:* can the feedback be noticed?
- *Unobstrusiveness:* does the feedback distract or annoy?
- *Minimal intrusiveness:* does the feedback compromise the privacy of others?
- *Fail-safety:* what happens if the user fails to take action?
- *Flexibility:* can it cope with different "comfort" levels?
- *Low-effort:* how much effort does it require on behalf of the user?
- *Meaningfulness:* does the feedback make sense to the user?
- *Learnability:* how natural is controlling privacy in the system?
- *Low cost:* is the deployment feasible?

The authors tested their guidelines on a feedback and control mechanism for a public reading and meeting area at EuroPARC. Their solution incorporated a large mannequin holding the video camera, thus providing a trustworthy, meaningful, and appropriately timed feedback mechanism for *capture*. A separate monitor would show the picture being transmitted, and optionally the names or images of the people currently subscribing to this video stream, though this might be intruding on the privacy of the watchers in order to provide *accessibility*. Both *construction* and *purpose* feedback was left unaddressed in the solution.

Having developed our set of requirements from an extensive analysis of ethics and legal guidelines, we arrive at feedback and control issues very similar to the user concerns at EuroPARC. Bellotti and Sellen list is in fact a subset of the Fair Information Practices presented in section 3.2.2 [32]. However, with our privacy contracts and the associated privacy proxies and databases we are able to provide *construction*, *accessibility* and *purpose* feedback in an unobstrusive, fail-safe, flexible, and low-effort manner: simply by carrying around our privacy assistant device, all data collections are unobstrusively logged and could optionally be compared to the individual privacy preferences of the user. Fail-safety is achieved for all optional services, which are only used if the user carries a privacy assistant, while mandatory services would need to provide some "real-world" announcement (e.g., a poster or prominent sticker) for legal reasons anyway. How "visible" the *capture* process is depends on the preferences of the user, who could set her privacy preferences such that her device alerts her, e.g., to all video recordings with a unobtrusive yet noticable vibration alarm.

Our proposed solution is thus very much compatible with Bellotti and Sellen's privacy guidelines, providing additional feedback and control mechanism for collaborative media spaces such as RAVE, where more direct awareness cues such as feedback monitors and embodied sensors are used. However, PawS is also applicable in situations with more unobtrusive, invisible sensors and services.

Privacy Lamps and Vampire Mirrors

Similar to Bellotti and Sellen, Butz et al. explore the concept of privacy in the domain of computer supported collaborative work, specifically when using immersive environments that replicate the collaborators rooms and desk at each end [53]. Using a virtual *privacy lamp*, a user can light up certain areas on the virtual representation of her desk that then mark a private area. Both virtual and real objects that fall within the light of the lamp are not replicated on the other side.

Another metaphor explored in their work is the concept of a *vampire mirror*, which acts like a mirror of one's desktop items, but leaves out images from objects that have been marked as private. This allows collaborators to quickly realize what part of their desktop is replicated to others.

As PawS focus more on data privacy rather than privacy of (real and virtual) objects, the directly applicability of this work is limited.

However, it makes a strong case for using everyday metaphors to both communicate and manipulate personal privacy levels.

Privacy for Home Media Spaces

Another similar approach is the Home Media Space Privacy project at the University of Galgary [252], where Neustaedter and Greenberg try to find everyday privacy feedback and control mechanism for a shared cooperative workspace in the realm of telecommuting. In contrast to the shared offices in EuroPARC, a shared audio and video environment that also doubles as a living room or bedroom runs a much higher risk of disclosing personal situations.

The authors use both explicit and implicit control elements, as well as audio and visual feedback, to provide telecommuters with privacy over their audio-visual collaboration link. For example, the camera can be manually turned on or off using an easily accessible (and easily identifiable) button, but also implicitly pauses its recording when the user leaves her chair. When changing back to recording again, the camera also audibly clicks and visually "twitches" in order to alert the user to the newly commencing recording. When being manually turned off, the camera not only stops transmitting images, but also swivels away from the user and faces the wall.

Implicit and explicit control are also realized in PawS, though not through direct manipulation of buttons but through the interaction with the user's privacy assistant. However, both control and feedback mechanisms can be significantly improved in special situations such as a telecommuters home office. All-purpose solutions such as PawS cannot make proper use of the specific affordances of such unique environments.

Privacy in Aware Homes

Several projects have investigated how families in remote locations can stay in touch through a "shared" living space inside each home.

The *Aware Home* research initiative (AHRI)[7] at the Georgia Institute of Technology has built a standard residential home that is outfitted with extensive network and sensing technology [251]. The *Digital Familiy Portrait* [250] is an augmented picture frame that provides not only a two-way intercom, but also awareness of the activity of the

[7] See `www.cc.gatech.edu/fce/ahri/`

remote family members. Instead of using intrusive surveillance technology such as video or audio streams from the remote location, the picture frame uses a series of butterfly-icons whose size represent the level of "activity" at the other location.

A similar sense of awareness using live video pictures was created as part of the *Interliving* project,[8] where a number of different communication interfaces, so-called *probes*, where deployed in the homes of participating families [171]. The *mirrorSpace* probe provides an interactive video communication system, yet offers not only feedback by overlaying the remote picture with a mirror image of one's own picture (as it is displayed on the other side), but also introduces the concept of *proximity* (c.f. section 3.4.4), whereas people further away from the mirror are blurred in the remote picture in order to provide their privacy [301].

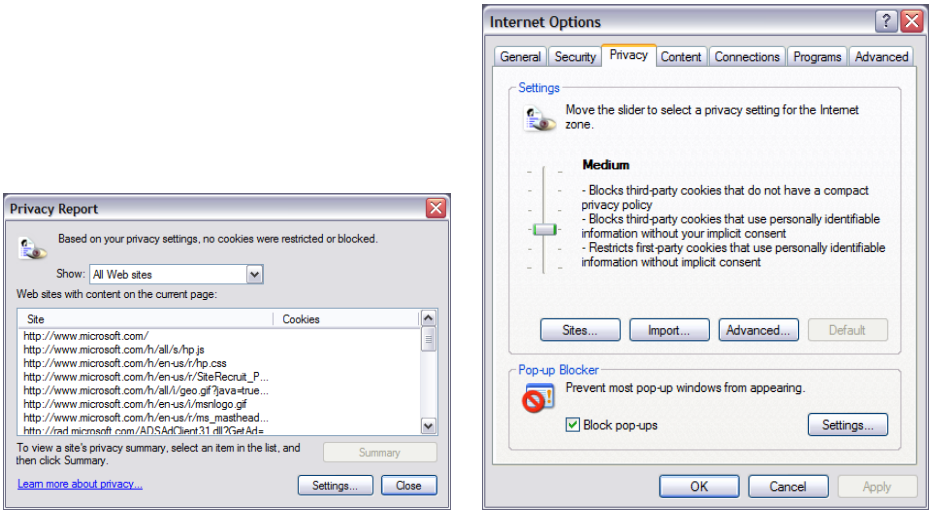
Privacy Mirrors

Privacy Mirrors is a framework by Nguyen and Mynatt for user interface design in the domain of ubiquitous computing privacy [253]. Based on prior research in environmental psychology, social translucent systems, media spaces, and privacy policies, the authors develop five characteristics that should be part of any ubiquitous computing system: *history*, *feedback*, *awareness*, *accountability*, and *change*.

Keeping a history of data flows and visualizing it to (i.e., providing feedback) allows data subjects to gain insights into the customs, norms, rules, and practices of their peers. Nguyen and Mynatt propose three different cognitive models to provide such data: glance, look, and interactive, corresponding to three different levels of detail. Awareness is an effect of history and feedback, allowing data subjects to understand how they "participate" in the overall system. Awareness, in turn, can create accountability, the "I know that you know" that can socially governs people's actions. Taken together, the data subject is thus able to perceive the overall system and can better anticipate how his or her actions influence it, enabling actively changing his or her behavior.

As the authors note themselves, many of their design guidelines have a direct correspondance in the Fair Information Practices, and thus are also present to some extend in the design of PawS. This allows their implementation in a dedicated PawS privacy assistant interface,

[8] See `interliving.kth.se`



(a) Privacy Summary (b) Setting Privacy Preferences

Figure 5.2: *P3P support in IE6.* The summary page shows all cookies set by the various referred pages, including a direct link to a policy summary using the "Summary" button. Privacy preferences – only regarding cookies – are set using a slider.

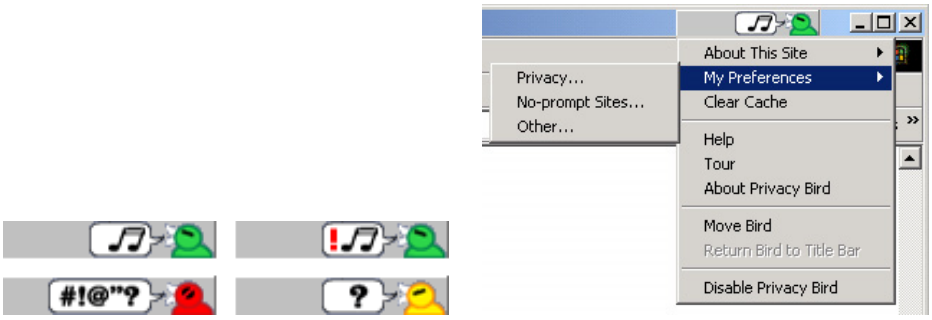
as the required information and methods are already part of the PawS infrastructure. In this way, PawS and the Privacy Mirror framework can be very well combined.

Web Privacy Assistants

The P3P initiative prompted a number of implementations that incorporated P3P into a Web browser, most notably Microsoft's Internet Explorer 6[9] and AT&T's *Privacy Bird*.[10]

Privacy controls in Internet Explorer 6 come as a six-level slider and only address the way the browser handles the placement of *HTTP cookies*.[11] User can choose from *no cookie control*, *low protection*, *medium protection*, *medium high protection*, *high protection*, and *block all cookies*. Advanced users can further configure these basic preferences. Cookies that have no or no compatible *compact P3P policy*[12] that describes their purpose will be blocked (i.e., not stored) by the browser. Internet Explorer 6 does not check for or operate on full P3P policies, though an inspection tool (see figure 5.2.a) allows users to display a human-readable version of a Web page's policy.

[9] See `www.microsoft.com`
[10] See `www.privacybird.com`
[11] See `en.wikipedia.org/wiki/HTTP_cookie`
[12] Compact P3P policies are one-line summaries of full P3P XML policies that apply to cookies and can be embedded directly in the HTTP headers accompanying the cookies [79].



(a) Privacy Status Icons (b) Menu Entries

Figure 5.3: *AT&T Privacy Bird UI.* Privacy Bird uses four icons to signal the compatibility of a site's privacy policy with the user's preferences (subfigure a), clockwise from upper left): compatible; compatible but with incompatible embedded content; no privacy policy found; and incompatible privacy policy. Preferences can be set through the added menu entries (subfigure b))

A much more detailed implementation is AT&T's Privacy Bird, which comes as an add-on to Internet Explorer 5.01 an up, and transparently analyzes Web site policies as the user connects and downloads pages. Comparing the (full) P3P policy to the user's preferences, it displays a colored bird icon in the browser's windowbar, indicating whether the policy matches the preferences or not (or whether there is a policy). Figure 5.3 shows the different icons displayed for missing, conflicting, and matching policies, as well as the menu items added to the browser interface.

Similar to the Internet Explorer interface, preferences in Privacy Bird can also be specified using a small number of high-level categories: Low, Medium, and High (see figure 5.4). However, the preference dialog also shows the implications of each setting directly below, making it easy for users to understand the different levels and making custom changes to them.

AT&T's Privacy Bird is probably the most comprehensive system for specifying user privacy preferences today. It makes full use of APPEL [80], the companion specification to P3P for formulating privacy preferences. As such, it provides a good example on how a corresponding full implementation of a privacy assistant for PawS might look like. However, the special requirements of mobility and ubiquitous service environments would still make a careful analysis of user interface requirement necessary.

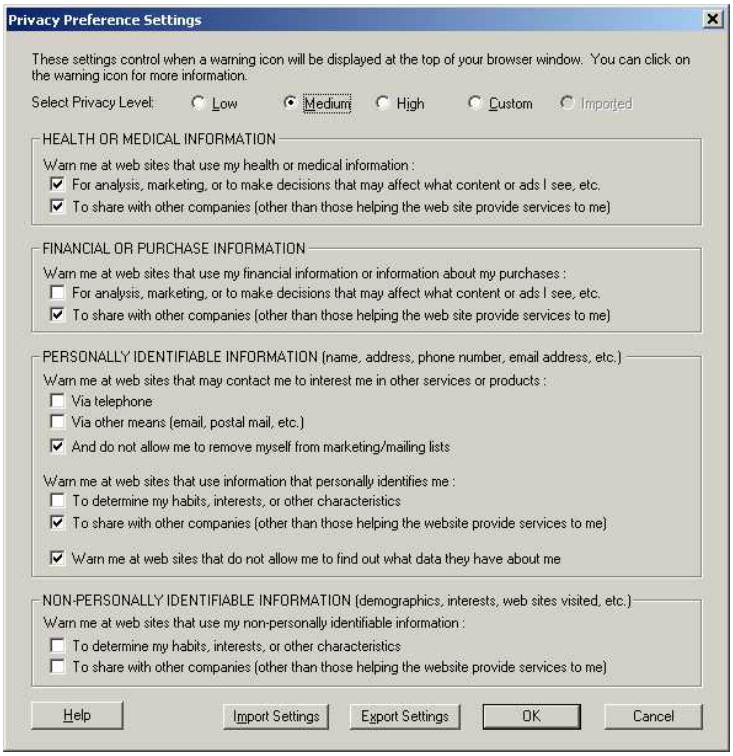


Figure 5.4: *Privacy Preferences in Privacy Bird.* Even though Privacy Bird also uses four high-level settings for privacy, users can immediately see the effect of each setting and easily make custom changes.

The Faces Metaphor

Probably the most applicable work related to our privacy assistant has been done by Lederer et al. at Berkeley [214, 216]. The authors specifically try to address the problem of giving the user an easy to understand metaphor for both assessing and influencing her current privacy level. To this extend they build upon the work by sociologist Erving Goffmann in the early 1960s, who studied individual identity and group behavior in his work *The Presentation of Self in Everyday Life* [138]. Seeing people as "actors" and interactions as "performances" shaped by environment and audience, Goffman constructs behaviors as giving a certain "impressions" that are consistent with the desired goals of the actor [26].

Lederer et al. similarly use a *faces* metaphor to ease privacy management for the individual. In their prototypes, privacy preferences are grouped into several "faces," each representing a number of dimensions such as what data to disclose and at what accuracy (e.g., location). Assigning specific situations to each of these faces, users can easily formulate rules such as "if a roommate makes a request while I am studying, show my Anonymous face" [216]. An example interface can

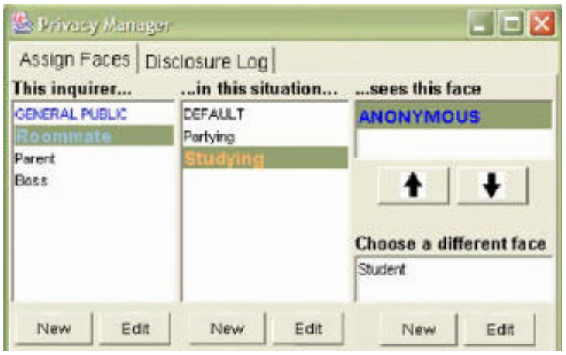


Figure 5.5: *User interface using the* faces *metaphor.* A roommate inquiring my status while I am studying sees my *Anonymous* face, which might entail not receiving my current location or other context [216].

be seen in figure 5.5. However, no implementation on a portable device exists, and the project has since been discontinued.[13]

Identity Management

Lederer et al.'s prototype is an instance of an *identity management* system, first proposed in the field of privacy technology by John Borking [43]. Borkings *Identity Protector* is a fusion of Goffman's work on roles and identity and anonymous certification technology: for each electronic interaction, a smart assistant (the Identitiy Protector) chooses a *pseudo-identity* from a subset of the user's real personal data, specifically avoiding to use *identifiable* user data (such as her name) and instead opting for anonymous certificates whenever possible. Herbert Burkert calls this "taking pressure off the consent principle" [52], as it allows data subjects to use services even without having to consent to disclosing their full identity.

The idea of identity management has since been explored in a number of projects, mostly in the area of Web privacy. The EU-funded PISA project [337], initiated by John Borking, aimed at realizing intelligent software agents for identity-protected information retrieval on the Web. A personal proxy approach is taken by the *DRiM* project[14] at the university of Dresden, Germany.

[13] Some of the collaborators have continued their work as part of the Confab project described in the previous section.

[14] See `drim.inf.tu-dresden.de`

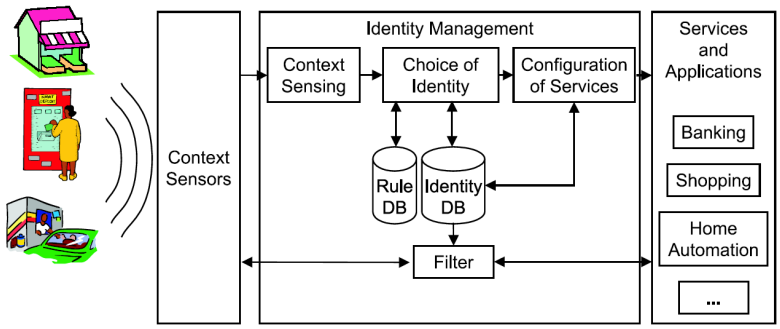


Figure 5.6: *The Freiburg Mobile Identity Manegement System* uses context to decide on the most apropriate identity in a given situation [181].

Freiburg Identity-Manager

The *Identity-Manager* System developed at the University of Freiburg is an implementation of Borking's Identity Protector concept in the realm of ubiquitous computing [181]. In contrast to Web solutions that identify the appropriate identity by the URL visited, the mobile identity management system at Freiburg University uses context sensors to choose the current user identity to present. Figure 5.6 shows an overview of the system components.

Based on previous experience with the Web-based identity management system ATUS [180], the system does not directly offer identities to the user to choose from, but instead presents her with a list of *tasks* that are apropriate for the current context, for which the various identities of the user are implicitly associated. For example, when being close to an ATM machine, the user can withdraw money using her bank identity (i.e., account number, identifier), while being close to a bus stop provides an anonymous timetable application [181].

The Freiburger model provides an interesting alternative to traditional privacy interfaces that force the user to select a specific identity. It also does not cover privacy policies and data management, providing a natural combination with PawS. Of course, the quality of such a user interface depends largely on its ability to properly recognize context.

5.1.3 Privacy Databases

The idea of combining data with metadata governing its use is already popular for enforcing digital copyright [44, 73, 329]. Successful implementation of this concept, however, requires use of so-called "trusted systems" [328] along the whole distribution chain, otherwise it would be fairly easy to separate data and metadata again. An alternative

approach is to provide pseudonymous, short-lived identifiers instead of the "real" data, thus incorporating a natural expiration time into the collected data [8, 128].

In contrast to digital media protection, however, privacy databases such as PawS typically do not aim for hacker-proof data protection but instead assume that the added-value of the system (i.e., having the system make sure that data collector honors privacy policy without costly manual verification) will make its usage popular among data collectors. Short-lived pseudonyms are also difficult to use for perception data such as a user's location, video picture, audio recording, or walking pattern.

Privacy databases are still in their early stages of research,[15] combining anonymization techniques[16] with policy management in order to provide both privacy-enabled storage as well as end-of-lifetime data anonymization.

Anonymous Data Mining

The goal of anonymous data mining is to develop accurate models without giving away access to precise information in individual data records. Latanya Sweeney's work at Carnegie Mellon University [331] focuses on queries over private data that return only k-anonymized data, i.e., none of the identifiers in the query appears less than k times in the result. Sweeney's protection model also takes into account the multiple-query problem, i.e., if anonymous queries over time are correlated through shared attributes to create identifiable data.

Agrawal and Srikant try to tackle this problem already one step earlier, at the time of data collection [13]. Their methods can be embedded, e.g., in browser plug-in's or other data collection system, and directly *distort* user-entered values so that the information stored in the databases does not allow recreating the exact value for an individual (e.g., the user's annual salary). Using their distortion algorithms, however, data collectors are still able to reconstruct the original *distribution* of values across the entire data collection, thus enabling sophisticated data analysis without needing to know the exact individual values. Their methods are limited to numeric attributes only, and still

[15] While some data warehousing systems, such as NCR's Teradata line of products (see `www.teradata.com`) are advertised to be "privacy-enabled," no detailed reviews or articles discussing its features are available.

[16] Database anonymization techniques build on prior work in statistical databases and multi-level databases, see [331].

require the user to be willing to disclose the attribute in the first place, even if it is a pertubed value.

Both principles could be readily applied in PawS, as both on the user interface side and at query time separate anonymization processed might be integrated into the system. However, Sweeney's k-anonymity represents a more general approach than Agrawal and Srikant's technique, which might not work well on perception data such as positioning information and does not work at all for symbolic information.

Hippocratic Databases

Agrawal et al. have also tried to incorporate policy-based privacy mechanisms into databases. Their paper *Hippocratic Databases* [9] sketches a system very similar to PawDB, in which privacy metadata governs data access. Just as the Hippocratic Oath has guided conduct of physicians for centuries, Agrawal et al. coin the term "Hippocratic database" for a system that "includes privacy as a central concern." The similarity between their proposed system and PawDB becomes apparent in the overall system layout, as shown in figure 5.7. In addition, Agrawal et al. also include preference matching (i.e., making sure that the user's preferences match the data collection policy) and query intrusion capabilities (i.e., detecting suspicious queries that are compatible policy-wise but do not match the regular queries from a particular department or user) in their system.

After the initial strawman architecture, Agrawal et al. have since subsequently implemented parts of their system, such as an XPath-based privacy preference language [12], P3P-based metadata control [10], and efficient query processing [11], gradually becoming a full-fledged system in contrast to the prototypical nature of PawDB.

Enterprise P3P

A more policy-oriented focus lies behind the *Enterprise P3P* (E-P3P) project at IBM Zurich Research. Karjoth et al. extend existing access control policy languages to allow inferences about full P3P policies [190, 192]. Instead of using P3P policies directly, however, they use a rule-based representation for defining valid recipients, purposes, and retention periods on a per-attribute basis, which is then translated into a P3P-compatible XML format for publication on a Web site. Rules might also define required operation for particular operations, e.g., a

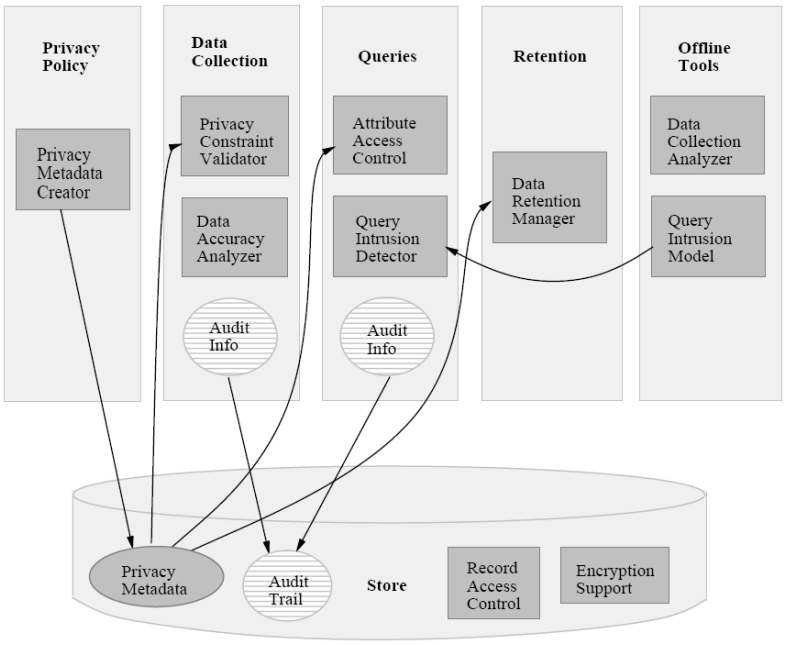


Figure 5.7: *The System Sketch of a Hippocratic Database* by Agrawal et al. [9] is very similar to the PawDB described in section 4.5.

notification of the data subject if the data is disclosed.

Instead of declaring a data usage policy in XML format, queries need to submit a fixed set of *context* attributes describing not only the query purpose and the party performing the query, but also the intended operation on the data, e.g., "read," "use," "disclose," or "anonymize." Karjoth et al. also suggest several extensions to the original P3P syntax in order to incorporate an improved consent model directly into XML policies [191]. Using this extended format, the authors are able to provide an efficient mechanism for evaluating P3P policies within an enterprise storage system, including transitory attributes such as the recipient or the retention period (which need to be updated as data and its metadata flow between different entities in a corporation).

E-P3P is a much more thorough approach to privacy metadata than what has been presented here in PawDB, which only aimed at demonstrating the feasibility of this idea in principle. Adding E-P3P to PawS would considerably strengthen its policy management features.

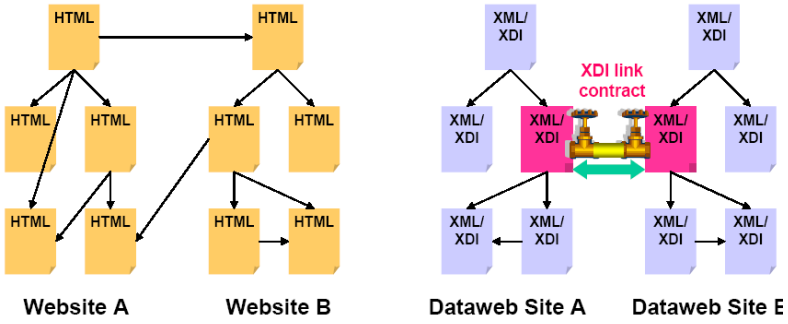


Figure 5.8: *The XDI/XRI Dataweb.* Instead of unidirectional hyperlinks, Constance's "Dataweb" uses *XDI link contracts* as bidirectional "pipes" that allow data access to be controlled [283].

XML-Policy Frameworks

A more Web-centric data management framework is the XDI/XRI initiative [17] by Cordance.[18] XDI ("XRI Data Interchange") is a protocol for exchanging resource links, specifically XRIs – eXtensible Resource Identifiers. XRIs are similar to URIs but are supposed to have a longer liftime than URIs.[19] A network of XRIs thus forms a "Dataweb," linking XML documents in the XDI format (which are called "Dataweb pages").

The XDI/XRI format grew out of an early version of P3P and thus uses a very similar apporach to data sharing: A Dataweb *link contract* describes (and controls) the flow of data between a Dataweb link, allowing two parties to share a certain piece of information (e.g., a calendar) yet retaining access control to that information through the active XDI link. Figure 5.8 illustrates this principle [283].

The XDI/XRI initiative is very similar to our *privacy contracts* approach, as both use bidirectional links to give users direct access over shared data. However, in contrast to PawS, the Dataweb framework does not actually *collect*, i.e., replicate personal information, but instead *links* to it using XDI/XRI. While this is an efficient solution for symbolic data such as business cards or calendar entries, it cannot handle sensory input that is directly collected and stored by the data collector.

[17] The work on XDI and XRI was originally called the XNS (eXtensible Name Service) project (see `www.xns.org`)

[18] Formerly *OneName*, see `www.cordance.net`. The intellectual property has been contributed to the non-profit OASIS consortium (see `www.oasis-open.org`) in order to encourage adoption of the standard.

[19] XRIs are either *i-names* or *i-numbers*. An i-name is a human readable identifier that maps to an i-number, which in turn is a machine-friendly identifier that is never reassigned (while the mapping between an i-name and an i-number might change).

5.1.4 Computational Trust

In contrast to the *social trust*[20] that PawS relies on, researchers in the field of network security have long used the term *computational trust* as a concept for decentralized access control systems, which compute whether a certain certificate holder is authorized for a specific transaction without relying on a central registry.[21]

This notion of *decentralized security evaluation* has made trust a popular concept for the field of ubiquitous computing as well, as such environments often tend to use distributed or peer-to-peer system architectures. In addition, ubiquitous computing systems are also expected to operate in a non-intrusive fashion, freeing the user from such banal things as usernames and passwords. Having identified trust as the vehicle for collaborating and socializing in a world without passwords and certificates (i.e., in real life), it seems to be the logical choice that we are simply to integrate its workings and mechanisms into our new security concepts for ubiquitous computing. By allowing our computerized agents to compute the "trustworthiness" of an electronic counterpart based on their local experience (and optional third party recommendations) directly on the spot, we could free them the tasks of soliciting and comparing access tokens (or privacy policies, such as in PawS) in a future world of intermittent disconnects and highly dynamic access patterns.

The following sections will review the current state-of-the-art in computational trust for ubiquitous computing in order to evaluate its merit as an alternative privacy model: instead of having to rely on *social trust*, merely "hoping" that a data collector will adhere to the posted privacy policy, would it be possible to *compute* the actual likelihood that the collector is stating the truth?

Notions of Trust

As trust based access control is gaining momentum in the field of ubiquitous computing, much remains unclear when it comes to defining the problem that such systems are trying to solve. In particular, work so far has often been confusing in terminology (even though – or maybe because – there is far from a shortage of definitions in disciplines such as philosophy, sociology, or psychology), vague on goals (other than

[20] See section 3.1.2.
[21] See section 3.3.1 for our discussion on certificates.

wanting to integrate trust into a system), and short of verifications (if you discount bar graphs plotting a developed formula). Our evaluation thus runs along three central questions: What problems are researchers planning to solve by incorporating trust into their systems? What kind of trust do they need for this (as there are many)? And how can they evaluate system performance during and after implementation?

Most trust-related projects in the area of distributed computing are trying to solve the problem of *certificate-based delegation*, that is, how to allow non-registered users through the use of previously issued certificates access to certain resources (e.g., [38, 54, 187]).

The mobile and autonomous agents community has used trust in order to *automate cooperation* between agents, e.g., for securing automated transactions such as shopping or job searching. This typically entails reasoning about the agent's intent, competence, availability, and promptness, rather than verifying a set of credentials [24].[22]

Research in wearable computing has focused on using trust to *automate transfer of personal information*, either by assessing the trustworthiness of the recipient ahead of time (e.g., [197, 307]) or by minimizing the amount of data exchanged until a certain number of (successful) interactions have taken place (e.g., [370]). While this could be seen as a specialized form of automated cooperation, the more informal nature of the data exchanged and its potentially high (personal) value often substantially change the requirements for such systems [371, 373].

Work in Web-commerce has recently begun to look into a very different problem of trust: that of getting *humans* to trust machines, not machines to trust machines.[23] Being more compatible with the notion of trust in the social sciences, they are analyzing the trust requirements of users in order to *raise acceptance* of e-commerce sites or shopping agents (e.g., [262, 291]).

While any of the above problems might be relevant in the context of ubiquitous computing (i.e., granting or denying access to certain services, using services in unknown environments, exchanging data with strangers, and creating acceptance for ubicomp in the community), most work in trust for ubiquitous computing has focused on expand-

[22] This was also the goal of Marsh [232], whose work has introduced many researchers in the area of ubicomp trust to the most prominent trust definitions from psychology, philosophy, and sociology. However, his work does not discriminate between the above reasons, e.g., intent vs. competence, but simply evaluates the payoff of different trust strategies in a given situation.

[23] In many cases, of course, we will need to trust the humans *behind* those machines, as [184] points out.

ing the trust concepts from network security, i.e., granting or denying access not simply based on pre-computed certificates, but rather depending on a particular context. While some projects simply try to incorporate generic context variables into the system (e.g., [317]), others explicitly base the computation on concepts from psychology, such as dispositional or situational trust and beliefs (e.g., [103, 316]). Their idea is to take established trust concepts within the social sciences and use them as a blueprint for building something different from "network trust" and more similar to "human trust" into their systems.

While the selected definitions might support sound theories in their respective disciplines, using them as the basis for computations is far from trivial. Especially total or even partial orderings over trust values pose serious problems for such solutions: Some designers envision humans to explicitly rate their trust in different people and situations [316], others stipulate an automated process to infer such values from observing real-world interactions [141]. Social scientists question whether explicit trust ratings based on questionnaires bear resemblance to actual behavior [158],[24] whereas deducing the level of trust through observation seems almost impossible, given the plethora of possible parameters that ultimately influence our (observable) decision to trust.[25]

Computing Social Trust

As we have seen in section 3.1.2, trust has become a rather fashionable research topic, not only in computer science but also in other (social science) disciplines. Computer scientists trying to reuse existing trust concepts as the basis for their computational framework can choose from a bewildering number of different facets and definitions of trust. This is not necessarily a good thing, as it not only shows that the concept of trust is far from clear (which increases the chance of picking the 'wrong' definition), but also significantly influences the system design due to the specialized nature of most of these definitions.

Hartmann notes in [158] that any definition of trust is always embedded into a theoretical framework that determines what can actually be

[24] Anybody who has ever tried to prioritize their (electronic) to-do-list will probably agree that the resulting order is in most cases only a very rough estimation of the real importance of each task, especially as new information theoretically requires a constant re-evaluation of priorities that few are willing to do.

[25] McKnight et al. [236], for example, list six sources that influence our decision to trust: trusting intentions, trusting behaviors, trusting beliefs, system trust, dispositional trust, and situational trust. A computer system would need to infer the composition of these input parameters given the observation of a single, binary output of "trusting" or "non-trusting" behavior.

explained with it. Definitions in the context of management studies for example try to explain (and improve) office workflows and group collaboration (e.g., [237]), psychologists use it to explain the formation of trust-relationships in families and friendships (e.g., [89]), and work in sociology looks at the larger context of trust and tries to explain its effects on communities (e.g., [224]), political systems (e.g., [255]) and economies (e.g., [129]).

This means that even though substantial work has been done in these disciplines that research in ubiquitous computing should take into account, simply picking one or more of these definitions as a starting point will in many cases not work, as the choice implicitly defines possible outcomes. This either leads to frameworks that are preceded by meaningless trust definitions, or produces trust architectures that mirror a process that is not applicable to the problem.

Good examples for such suboptimal transfers of concepts might be the history of flight (where imitating the mechanics of birds failed to get people into the air) or the development of world-class chess computers (which once were thought to represent the ultimate proof of artificial intelligence, yet turn out to best work using brute-force searching algorithms). In the context of trust, this for example results in frameworks that end up with so many variables that could potentially affect a single trust decision,[26] that neither explicit solicitation nor implicit learning seems possible. In a similar fashion, some systems have taken research on social networks [347] and – assuming trust transitivity – envision that one will automatically trust people that our close friends trust in turn [137, 142], even though there are plenty of examples where two close friends of us do not get along at all (but who would be required to trust each other due to their common trust in ourselves).[27]

Validating Trust

One important yet often overlooked aspect is the validation of the system. Few work on trust in ubiquitous computing has actually tried to verify the proposed solutions.[28] Instead, a framework's flexibility [317]

[26] [141] for example computes trust out of values for dispositional trust, situational trust, system trust, trusting beliefs, belief formations, and trusting intentions, each in turn representing a individually customizable context-dependent function.

[27] Consequently, Marsh [232] defines trust to be non-transitive.

[28] [141] uses a simulated game of blackjack to verify the framework, though its high abstraction level (only one player and one dealer, the player either always pays his debts, randomly pays, or never pays) and high level of customization (all parameters are adjusted to fit the desired outcome) limit its applicability to other situations.

and/or its similarities to psychological concepts [103, 316] are often cited as proof of its power. A reason for such shortcomings is certainly the above mentioned vagueness of trust-based ubiquitous computing system with respect to their goals: if the actual reasons for incorporating trust into a system are not made explicit, any kind of validation becomes impossible.

As trust is certainly a complex issue, validating systems that attempt to incorporate human trust might be far from trivial. A similar problem in the field of artificial intelligence (which undoubtedly produces complex systems as well) had been solved by introducing an indirect testing strategy: rather then setting a specific task to solve (such as solving a puzzle, or playing a game of chess), the Turing-Test asks humans to judge whether a conversation partner is actually a fellow human or just a computer trying to pose as one (interactions are properly disguised through non-verbal and delayed communications).

Designing a similar testing scenario for evaluating the effectiveness of a trust-based ubiquitous infrastructure could thus involve a range of automated trust-frameworks that would compete with human "trust assessors." If a statistically relevant number of observers could be tricked into believing that trust decisions taken by a computer system were made by a human, the corresponding trust-framework would have passed the "Trusting-Test." It remains questionable, however, if an observer would not be equally likely to identify a completely random system (or a very simple one, e.g., featuring a "tit-for-tat" strategy) as being "human", simply because the plethora of reasons that could influence such a decision might make even random decisions look somehow "believable."

A more useful test would probably be to compare the system's decisions to our own, personal decisions regarding trust (e.g., whether we would buy our concert tickets from the same ubicomp services that our system would). Again, judging the outcome of such a test would be difficult. Maybe the system did not possess enough information in order to reach the same conclusion as we did? Maybe the situations where we disagreed were really split decisions that could have just as likely gone the other way? Whatever the overlap between our choices and that of the system might be: The "usefulness" of such a system would probably depend largely on the subjective attitudes of each user (i.e., how much "leeway" they are willing to tolerate), rather than actual system performance.

One solution might be to maximize the system's performance in absolute figures, rather than with respect to personal preferences. So instead of trying to *emulate* our behavior, we would build a system that would try to *improve* our behavior, given an optimal outcome for each situation. As most of the proposed ubicomp trust-frameworks require a comprehensive risk-assessment[29] in order to correctly compute their trust values, calculating the benefits between two different trust strategies (the one of the system vs. my personal decisions) would in theory be feasible (even though the initial risk assessment might not).

Taken altogether, the problems associated with validating trust systems could indicate a fundamental incompatibility between our "human" notion of trust and the computational processes that try to mirror them. Even if such systems would ever get enough data through user solicitation or observation, we might only be able to judge their performance with a few toy examples: Since any serious trust-based decision could potentially allow any number of arguments to be made for any number of outcomes (i.e., whether to trust or not), who are we to say that the system made a mistake (maybe *we* did)? And should the system ever get it wrong (by whichever standard), it might simply indicate a lack of consistency on behalf of the user (who fed conflicting information into the system), rather than a system design problem.[30]

Applicability to PawS

Three important aspects are often missing from today's trust-frameworks in ubiquitous computing. The lack of (good) scenarios exemplifies the often ad-hoc implementation of trust into the infrastructure, which also hinders the selection of the proper trust models to use. The currently developed solutions consequently make validation seem impossible, simply because the authors never describe what constitutes a successful operation of the system.

Should the above questions be thoroughly answered in existing and future frameworks, it might become clearer which goals can and cannot be solved by incorporating certain notions of trust into computational frameworks. Given the described difficulties associated with validating a system that assesses the trustworthiness of strangers for us and engages in collaborations with them on our behalf, it remains question-

[29] I.e., how much am I to loose if the other does not do what I trust him to do

[30] A comparable endeavor might be the construction of a computerized art critic that should judge the value of a painting or sculpture.

able whether any form of solicitation or implicit learning will ever be able to completely grasp the complexity of human trust. While specialized solutions to very specific problems might be able to benefit from a very restricted, computational notion of trust,[31] any generalized solution might work better by *supporting* the human trust-based decision process (i.e., by providing relevant information on demand but leaving it to the individual's state of mind whether to trust or not) instead of trying to *mimic* it. This, in essence, is the approach we have been taking in our PawS infrastructure.

5.1.5 Summary

This section has looked at a range of related tools in the area of ubiquitous computing privacy. Both the work by Myles et al. [249] and the Confab project [167] are very similar to our PawS privacy infrastructure, using metadata to govern data collection and usage. Both try to address location information more explicitly than we did in PawS, something that we will discuss in section 5.2 below. Apart from this focus, however, they follow the same design guidelines and principles that we outlined in our chapter 3: instead of providing bullet-proof security, they also rely on legal and social tools to enforce privacy promises. While the PAW project [56] also attempts to provide a comprehensive privacy protection framework, it is still in its conception phase and is thus difficult to compare.

PawS does not address user interface aspects – it merely uses a simple informational layout in its privacy assistant to demonstrate the function of the deployed privacy beacons (see section 4.4). While some projects more explicitly focus on the design of such a privacy assistant (e.g., [214, 180]), their applicability in ubiquitous computing environments have often only been tested on a single example application. It also remains a challenge how to consolidate feedback and control mechanisms outside a traditional screen interface, as suggested by the work of Bellotti [32] or Neustaedter and Greenberg [252].

Our work on PawDB has preceeded the recent activity within the database and data mining community on *hippocratic databases* (to use the term by Agrawal et al. [9]). While the efficiency of anonymization techniques such as Sweeney's k-anonymity [331] (and even more so

[31] An example of this would be traditional trust-management in computer networks, as exemplified by [38, 187].

Agrawal and Srikant's work [13]) has not yet been applied to non-traditional data elements such as perception data, it nevertheless remains an important aspect in any privacy-preserving storage concept.

The alternative of actually *computing* trust rather than relying on the effect of social trust remains doubtful. Research in the field is still in its infancy and will require a much clearer set of requirements and evaluation principles, before useful mechanisms for ubiquitous computing privacy might be incorporated into privacy frameworks such as PawS.

Taken together, our work on PawS interfaces well with a number of alternative approaches. By having our set of requirements grounded in a thorough analysis of the social and legal realities of privacy, we have been able to identify a broadly shared consensus on how it is that we might protect our privacy in future ubiquitous computing environments.

5.2 Location Privacy Tools

Location privacy has recently gained increased attention from citizens, lawmakers, and researchers alike. Driving forces are the spread of location aware services such as friend finders[32] and location-aware emergency services such as the E911 mandate in the US.[33]

At first sight, location data is just another piece of a person's informational privacy, similar to a person's name, home address, or profession. However, knowing a person's location at a specific point in time often allows a substantial number of inferences to be drawn, e.g., regarding his or her hobbies, friends, political inclinations, or even sexual preferences. The following sections will try to analyze the issue of location privacy in more detail, including how such information is collected, potentially used (legaly and illegaly), and what technical proposals exist

[32] In Germany, MECOMO AG offers the *FRIENDS.nextome* service for user of the E-Plus and O2 networks, which allows tracking the current cell-location of mobile phones (see `www.mecomo.com/friends.nextome/`). Similar services are Mapion's *Imadoko* service in Japan (see `imadoko.mapion.co.jp`) and VeriLocation's tracking service in the UK (see `www.verilocation.com`).

[33] E911, or *Enhanced 911*, is a service mandated by the Federal Communication Commission (FCC) in the US, requiring all telecommunications operators to associate a physical address with the calling party's telephone number in case of an emergency call [356]. While this can easily be implemented for residential lines by a reverse lookup on the phone number, phase two of the E911 program requires mobile phone operators to provide a similar level of detail. Cell data alone will not be sufficient for this precision, as callers must be located within 50 to 300 meters (see `www.fcc.gov/911/enhanced/`), requiring special location hardware in mobile phone base stations. Implementation of E911 phase two must be completed by December 31st, 2005.

to protect location information. It concludes with an assessment of both the proposed solutions and our own approach taken with PawS.

5.2.1 Collecting Location Information

There exists a wide variety of location systems, with different localization methods, positioning precision, deployment and maintenance costs, and application areas [164]. However, besides the *How?* of localization, also the *Why?* is important. This is because location information is not only collected intentionally for the fulfilment of a (location-based) service, but also often as a by-product of using localized (but not location-based) services:

- *Localized services* (location as a by-product): Location information is increasingly available in electronic services that are used while being on-the-move, even though the location information itself is not necessary for performing the service. Well-known examples are mobile phones[34] or credit cards,[35] but potentially any service that uses non-remote electronic user interaction (in contrast to Web shopping) can compromise a customer's location privacy as it links a user's action to a particular location.
- *Location-based services* (location as an attribute): Often seen as the killer applications of ubiquitous computing, location-based services explicitly operate on a customer's current or past location(s), e.g., for finding close-by restaurants, automatically calculating road or train fares, or hailing a taxi-cab. These in turn can be subdivided according to the duration of the location disclosure:
  - *On-demand:* A single automatic or user-initiated transfer of location information is necessary to use the service, e.g., calling a cab to the current location, dialing an emergency service, or finding a list of restaurants in the area.
  - *Tracking:* The user's location is disclosed over a period of time, e.g., a fleet management system providing a real time view of the location of all company cars; an active badge-like system that tracks office workers, hospital patients, or convicts

[34] While technically the location of a mobile phone is very relevant for providing reachability, this is rather a technical limitation due to the limited cell size. A satellite phone can provide similar communication services without disclosing a user's location on a cell-level.

[35] Automated teller machines (ATMs) also provide banks with information about the customers movements, though banks have so far explicitly refrained from using such data [254].

on probation; or a friend finder that shows me the location of nearby colleagues and friends.

5.2.2 Privacy Threats

Before examining the proposed (technical) solutions for location privacy, we briefly want to examine the exact nature of the threats posed by such systems.

Attack Models

Possible attackers in the area of location privacy are no different from those in "regular" privacy areas, though the more sensitive nature of location information, e.g., when compared to a street address or a person's age, raises the possibility of such attacks.

- *Individual attackers:* Neighbors, friends, or family might be interested for personal reasons to know one's current or past location. This also includes criminals that would use this information to plan a break-in, robbery, or assault.
- *Malicious companies:* Greedy corporations might decide to ignore an agreed-upon contract and use customer location data for other than the agreed-upon purpose, share it with unintended parties, or store data longer than allowed. Companies might also not offer such promises in the first place, or lure the customer into providing such data for a small financial incentive. This does not only include supermarkets and retail chains, but also insurers or potential employees.
- *Law enforcement:* In order to find known criminials, potential suspects, illegal immigrants, or parking violators, law enforcement agencies might legally access collected location information. This also applies to a defending party in a civil lawsuite, e.g., in a divorce.

While the above list focuses exclusively on *disclosure* attacks, i.e., when data of an individual gets disclosed to a party that the user wanted it to be kept private from, other attacks are also relevant in the context of location privacy, specifically denial-of-service attacks (i.e., preventing location information from being distributed) and integrity

attacks (i.e., changing the reported location of an entity from its true location) [86].

These parties have several possibilities of obtaining an individual's location information from a location infrastructure:[36]

- *Information Leakage – Position:* A service receives more detailed location information than necessary, e.g., calling a cab to my office not only reveals the street address but my exact room location I am calling from.
- *Information Leakage – Time:* A service receives information about the times I have been at a particular place, even though it does not need it to provide its service, e.g., a road toll system that records the exact entry and exit times of each vehicle, thus enabling police to give out speeding tickets if the distance has been travelled in too short of a time.
- *Information Leakage – Identity:* A service learns the individual's real identity although it could have been used with a pseudonym or even anonymously.
- *Collusion:* Location data from two or more services (or two or more independent datasets from the same service) are merged in order to gain additional information, such as the user's identity or movements over time or more precise location.
- *Eavesdropping and Trespassing:* An attacker listens in into the communication between parts of the location system, or breaks into data stored on a location server, in order to learn the current or past position of an individual.

Eavesdropping and trespassing (i.e., unauthorized access of stored information) can be restricted using relatively straightforward mechanisms (encrypted communication, sender and receiver authentication, etc. See section 3.3.1 on page 84), the main issues for location privacy are thus preventing the unintended collection of unnecessary information (data minimization principle) and the post-hoc or real-time collusion of several anonymous information sources. Especially the combination of anonymous location data is a challenging problem.

[36] The IETF working group on Geoprivacy groups these attacks into three classes instead: protocol attacks, host attacks, and usage attacks [86]. The Geopriv initiative focuses explicitly on security mechanisms to prevent protocol and host attacks, and on privacy rules to prevent usage attacks. See section 5.2.3.

Data Combination

As the work by Sweeney [331] has shown (see section 5.1.3 on page 196), even anonymously collected information can easily combined later to form identifiable information.

Rodden et al. [295] remark that disassociating already one attribute of a (Location, Identity, Time) tuple provides strong location privacy. However, as the work by Zugenmaier et al. [383] points out, even disassociated datasets can be subject to an intersection attack by a determined attacker. By using Zugenmaier et al.'s *Freiburg Privacy Diamond*, the likelihood of associating a certain action at a certain location and time to a certain user can be computed, providing an important tool for analyzing proposed anonymization techniques for location privacy.

Similar work has been done by Beresford and Stajano [35, 36], who analyse the use of pseudonyms in location systems. Using simple heuristics, such as a person's office and which pseudonym spends the most time at this desk, Beresford and Stajano were able to correctly deanonymize *all* pseudonymized users in a location system deployed in their lab. The authors propose to designate special *mix zones*, similar to the mix nodes in a mix network (cf. section 3.3.2 on page 89), in which no application requires location updates and which can thus be used to "mix" pseudonyms of all users currently located in this area.

5.2.3 Proposed Solutions

Assuming end-to-end security between position sensors (i.e., entities that report an individual's current position) and location servers, as well as data storage security, proposed solutions focus on regulating legitimate access, preventing collusions attacks, as well as minimizing information leakage by carefully selecting the type of information to transmit and store.

Two trivial solutions exists for both localized and location-based services, given a particular, restricted application model: anonymous localized services, and non-dynamic location-based services.

Localized services that do not need the identity of a user can simply operate anonymously. For example, a store that wants to inform customers in its vicinity that a sale is going on can use anonymous broadcasts; a train company that wants to monitor the number of people on a train platform (to prevent overcrowding) might count the number of

train passes it "sees," yet would not retain individual IDs; an emergency system for locating office workers in case of a fire would track badges but not their IDs. However, simply not storing identifiable information might run counter to security and safety concerns of both society in general and the service provider in particular.[37] Also, as pointed out in our analysis (see section 2.2 on page 36), real-world data is often hard to anonymize, leading to identifiable information even with unidentifiable raw data, simply by correlating it with additional information.

As described in section 2.1.2 regarding personal borders, any information crossing unexpectedly physical, social, spatial, or temporal borders can constitute a privacy invasion, albeit on a more instantaneous scale. Imagine a heat- and movement-sensor based emergency system in a hotel that anonymously tracks the location of guests in the hotel, in order to guide rescue workers. Overattentive hotel employees might use this information to check on the availability of guests in their rooms, e.g., whether they are currently in the bathroom when a phone call comes in. While these are valid privacy concerns that are potentially aggravated through the deployment of location sensing technology (even if used anonymously), solutions are often to be found in the social and operational realm, rather than looking for better technology.

Similarly, a location-based service that does not need dynamic third pary information, e.g., a map application, can use self-positioning and local lookups (e.g., on a CD) to keep the user's location and his or her information needs (e.g., where is the closest restaurant or movie theater) private. Again, while technically feasible, economic constraints might render such systems less attractive in the future. Both mobile communications providers and location-based service companies have an incentive to encourage the user to dynamically submit his or her current location to a remote application server whenever the application is used. Also, users might profit from more current information and lower service costs (as the dynamic usage model might be offered at a substantially lower rate than providing the entire dataset to the user ahead of time).

Technical solutions thus have to focus on the remaining cases:

- *Location obfuscation:* Instead of asking service providers to ignore identification data, tracking systems might be setup so that the true location of a user remains unknown within certain parameters.

[37] Even strict European privacy legislation allows for exceptions to the *data minimization* rule in the interest of safety and security.

- *Identity obfuscation:* By dynamically assigning pseudonyms instead of real identities, location systems can make it difficult for a privacy-invasive third party to infer the real identity of a tracked user.

- *Access control:* For all legitimate cases of location disclosure, technology can provide mechanisms to selectively allow or disallow others to see a user's current location.

Proxy-Based Identity Protection

One of the first location systems, the Active Badge system at Xerox's Palo Alto Research Center, already addressed privacy issues using so-called *Location Query Services* and *User Agents* [324]. Each User Agent collected and subsequently controlled access to all personal information of it's owner, including the user's current and past locations. To obtain a user's location, User Agents would register their owner's ID with all available location servers, prompting them to notify them whenever the badge with the corresponding ID had been sighted.[38] Each badge would use a pseudonymous ID that would only be known to it's owners User Agent,[39] thus forcing queries for the current location of a known user to address the user's User Agent, where regular access control mechanisms can control disemmination of the user's real identity.

Applications starting from a known location instead of a known user ID that tried to obtain a list of current user's at that location, would use a Location Query Service (LQS). An LQS manages a list of object tuples, containing a location identifier, an RPC handle, and an optional attribute list, describing for example the type of the object. While public resources such as printers or displays would register their full list of attributes with an LQS, a User Agent could decide whether or not it would register itself with the full list of attributes (e.g., the user's ID), with only an anonymous handle (i.e., providing only a location ID and an RPC handle, but no attributes), or not to register at all.[40] A query for a certain location would thus return a list of tuples with varying detail, prompting an application to contact individual User Agents

[38] Unix terminals would do so as well whenever a user would physically login.

[39] However, see Bereseford et al. [36] for details on how fixed pseudonyms can easily be correlated to real identities by using real-world constraints such as a person's office.

[40] User Agents could even register multiple identities for the same user in order to hinder traffic analysis.

through the given RPC address for more information, thus allowing User Agents to implement access control to the user's location.[41]

Rodden et al. [295] additionally introduce encryption into such a proxy-based approach in order to facilitate group disclosures. Instead of providing a resolvable RPC address, the user's User Agent stores a self-chosen pseudonym that it encrypts with the receiver's public key[42] and labels with the receiver's ID.[43] By notifying the receiving application of the chosen pseudonym, only this application can relate a particular piece of location information at the location server to this particular user, and only as long as the user continues to use that pseudonym. Using an encrypted pseudonym instead of storing the chosen pseudonym directly allows managing group subscriptions: while the User Agent uses only a single pseudonym for a single tracking service, even if a number of parties subscribe to it,[44] the location server can store this in a larger number of individual information pieces – one for each subscribed party. Should the user decide to remove or add a party, it's encryption key can simply be added or removed from this list of information pieces. Similar schemes have been developed by Hauser and Kabatnik [159] as part of the NEXUS project,[45] and by Kesdogan et al. [194] in the context of the GSM network.

Rule-Based Access Control

In contrast to the identity protection systems described above, systems based on rule-based access control focus on controlling the circumstances of data release, rather than hiding user identity behind a pseudonym. Such systems typically build upon traditional access control systems that have been extended with location specific features.

For example, the Houdini system at Bell Labs [170] uses a classical access control list in the form of resource-user pairs to control whom location information gets disclosed to. However, resources are called *contexts* and apply to the user's current location (e.g., at home, in the office, or in a shop), while user access rights can additionally be constrained using realtive distance between users. For examples, rules in

[41] RPC handles might additionally use proxies or mix networks to hide the network address of a User Agent.

[42] Or any other key agreed-upon with the receiver of the location data.

[43] This could, for example, be the hash of the recipient's public key itself.

[44] An example for such a multi-subscriber service would be a Friend Finder service, where a user shares his or her location with a number of friends.

[45] See `www.nexus.uni-stuttgart.de/`

Houdini can allow familiy members to see me while shopping, whenever they are within a 10 mile radius from my current location. The individual rules and each user's current location is stored in a central, trusted location system in order to be able to compare relative distances of users.

Hengartner and Steenkiste [162] specifically address location privacy as part of the Aura project (see section 5.1.1 above), allowing users to formulate location disclosure rules that operate not only on the identity of the requestor but also on the current time and the current location of the user. Additionally, rules can explicitly set the granularity level of the return location information. Aura also supports *room policies* that allow room owners to override the policies of the users currently in the room (e.g., a user can always find out who is in her room, even if the user's own policy would not allow a location disclosure).

With a more traditional network security oriented focus, neither Houdini nor the Aura system support preferences involving the purpose, recipient, or retention aspects of a data collection, as P3P-based systems like PawS or the system by Myles et al. [249] (see section 5.1.1). Aura's room policies are implicitly present in PawS as *mandatory* data collections, albeit at a lower granularity.

The Geopriv Working Group

The IETF working group on Geographic Location and Privacy (Geopriv)[46] tries to define a both architecture and protocol independent model for accessing location information. The Geopriv requirements document [85] defines the following principal entities:

- *A Target* is the person or object whose location is to be communicated.
- *A Viewer* is the final recipient of the information about the Target's location.
- *The Location Generator (LG)* initially determines the location of the Target and creates Location Objects that describe the Target's location.
- *A Location Object (LO)* conveys location information and optionally privacy rules that are processed by Geopriv security and privacy mechanisms.

[46] See `www.ietf.org/html.charters/geopriv-charter.html`

- *A Location Server (LS)* stores Location Objects created by Location Generators. It is responsible for applying the rules of the Location Object.
- *The Location Recipient (LR)* is the entity that receives the Target's location information, either on a query-by-query basis, or through a long-standing submission, in order to provide it to the Viewer.

Location disclosure rules in the Geopriv framework operate on specific geographic conditions, namely if the user is within a symbolic location (e.g., at a street address or in a city), within a geospatial location (i.e., a polygon defined by longitude and latitutde coordinates), or within a specified altitude. Rules can specify *actions* and *transformations* that must be executed by a Location Server. Actions are application dependent and are not part of the Geopriv framework [311]. Transformations specify if a Location Server may distribute the Location Object, how it must change the level of location detail if it does so, how long it may retain that information, and if the rule information must be deleted when forwarding a location object to another recipient.

In contrast to our work on PawS, purpose information is not used in the Geopriv framework. Its main focus is setting forth security requirements for a location infrastructure, and providing means to scope rules by time and space.

Self-Positioning Systems

A popular alternative to a centralized location systems are architectures that support self-positioning, such as the ubiquitous GPS system.[47] Proponents of such wearable systems often declare the privacy problem being solved by having the user locate herself, instead of having to trust a central infrastructure [290]. As GPS can only be used in an outdoor environment,[48] numerous alternative self-positioning systems for indoor navigation have been devloped.

The *Cricket* system [272] uses an array of beacons mounted on the ceiling that emit both ultrasonic pulses and radio signals. Users carry mobile receivers that can detect the radio transmissions and measure

[47] GPS – the *Global Positioning System* – is a satellite-based outdoor navigation system operated by the US Department of Defense. It can be used by anyone, free of charge [366].

[48] GPS receivers require signals from at least four satellites in order to compute their longitude, latitude, and altitude. Signal reception is disrupted by tall buildings in cities, and by walls when used indoors [366].

the difference in time-of-arrival to the corresponding ultrasonic signal of the same beacon. In order to lower the probability that signals from two beacons overlap, signals are sent only four times each second. Also, to minimize the effects of reflections and environmental ultrasonic noise, receivers average over multiple samples. This lengthens the time needed for a single location update to up to five seconds. Randell and Muller [278] use a single radio signal from a base station to trigger a series of ultrasonic pulses from several beacons in a predefined order, which allows receivers to update their position up to 10 times a second. Hazas and Ward [160] further improve performance of such systems (to up to 40 position updates per second) by using broadband ultrasonic pulses that are able to carry identification information from each beacon, thus alleviating the need for Randell and Muller's round-robin scheduling.

While self-positioning works well for static information systems such as map applications, the need for data exchange with other users or services renders the advantages of such a system often void. Solutions for location privacy that rely on self-positioning alone are thus comparable with solving information privacy through strong encryption mechanism: while they allow to keep personal information private, they do not help in those cases when a data exchange is explicitly needed.

Location obfuscation

While some of the above rule-based systems already provide for adjusting location data to a user-defined granularity (e.g., returning only the city name instead of the exact street address), some research focuses on building systems that dynamically alter the location detail of a user based on general system properties.

Gruteser et al. [147] assume a sensor network deployed in a single office building or even across a whole city that tracks an individual position while hiding her true location among a set of $k-1$ other subjects, calling this k-anonymity (as inspired by Sweeney [331]). Using a hierarchy of location controllers, e.g., from room controllers over to floor, building, and city level controllers, the network pertubes the sensed location of an individual in such a way that $k-1$ other indiviudals are in the same reported location. For example, during business hours, the location of an office worker might be reported at floor-level precision, while after hours only the building code is reported.

Gruteser and Grunwald apply k-anonymity to a central location ano-

nymization system [145], where detection events and user messages[49] are collected, obfuscated to provide the desired level of k-anonymity, and routed through a mix network in order to prevent traffic analysis attacks. This architecture also allows introducing delay and time inaccuracies for applications that do not need immediate responses, by keeping the location information accurate but witholding the position update event until $k-1$ other individuals have also passed through the area. In a follow-up work, Gruteser and Liu [146] extend this work to create k-area anonymity, i.e., individuals can define sensitive regions for which the system hides their true location within at least $k-1$ other sensitive areas.

The big advantage of this approach over rule-based obfuscation systems such as Hull et al.'s Houdini system lies in the improved usability: individuals do not have to manually set up rules or decide on a case-by-case basis what level of granularity is most appropriate under certain conditions, but instead specify a single number k indicating the level of desired anonymity. Such an approach could also be combined with PawS, as it leaves it to the service provider to specify the level of detail required for a certain application.

5.2.4 Summary

As more and more location-based services are beginning to appear in the marketplace, location privacy will play an increasing role in our daily life.

The principle of *data minimization* from the Fair Information Practices (see section 3.2.2 above) becomes paramount for location systems, as data combination techniques can quickly create comprehensive movement profiles if unnecessary information is collected and subsequently stored.

This applies especially at the *collection* level, where architectures like Bereseford and Stajano's *mix zones* [36] or Gruteser et al.'s k-anonymity [145] try to minimize the amount of location and/or identity information disclosed to applications that are able to operate with pseudonymous or reduced spatial/temporal precision. Proxy-based pseudonymity services such as the systems introduced by Rodden et al. [295] or Hauser and Kabatnik [159] are able to hide the true identity of a user

[49] Sending a message from a mobile device constitutes using a *localized* service, i.e., location information is implicitly present in an otherwise non-location-dependant service.

behind a long- or short-lived pseudonym, but fail to address the vulnerability of pseudonymous information to corellation attacks that was pointed out by Beresford and Stajano [35]. Also, future location-based services will most likely require personal user accounts or payment via credit card, thus rendering anonymity ineffective in such cases.

Self-positioning systems like GPS or indoor-systems like Cricket [272] are often seen as a solution to the overcollection of personal data, as they do not leak any information to third parties in order to position the user. However, as soon as the user wants to use dynamic information from a service provider, location information will need to be disclosed. Given current economic trends towards online pay-per-use services [39], relying on locally available information alone might prove to be too expensive[50] or not accurate enough.

Thus no matter whether location information is obtained through a positioning service or self-positioning, location privacy needs also be addressed at the *application* level. While a number of rule-based approaches exist (e.g., [85, 161, 170]), they often only extend existing access control framework with location parameters in order to allow access rules that apply to a particular geographic region. P3P-based approaches like PawS in comparison use preferences that operate on purposes, recipients, and retention declarations. While such preferences could obviously extended with geographic matching as well, a more suitable approach within these frameworks would be to embed such parameters into the collection policy instead. An office-awareness system, for example, would thus only operate using sensors installed within the companies buildings, or alternatively, subscribe to location information from a location provider (e.g., the user's mobile phone operator) using a spatial and temporal subscription format.

Location-based services would thus seem to be pose no new legal challenges, compared to any existing service that uses customer data. In the context of the EU Directive [94], the same regulations regarding the purpose, recipient, and retention declarations apply equally to conventional information such as a customer's address, and to dynamic information such as the customer's current location. A restaurant guide or taxi service would need to delete data about a user's location as soon as it is not needed for billing purposes. Using a central location collector such as one's mobile phone operator and selectively subscribing to

[50] As service provider price their online offers more competitively than, say, a CD-Rom version containing their entire database.

third-party services that the operator subcontracts would even allow a completely anonymous usage model, as payment for such services could be handled transparently by the location collector, making an explicit identification to the third-party service unnecessary.

For services that do not need location information but which nevertheless generate location information (i.e., *localized* services), both mobile phone networks and credit cards have provided long-running examples of how such information is to be handled. Frameworks such as the EU Directive regulate how such data must be processed, which typically only includes the original purpose for what it is collected, and storing it only for as long as such data is necessary, e.g., for billing purposes.[51]

From a technical point of view, location privacy can thus be supported by two things: anonymous location infrastructures that allow anonymous usage of location-based services in the first place, and transparency protocols that allow customers to understand and decide how their data is collected and used by a service. The challenge of anonymous location services lies in the high potential of data mining, as even anonymized data can often be correlated using heuristics such as correlating a person's personal belongings or often-visited places, such as his or her office. The challenge of transparency protocols lies in the need for managing potentially very frequent data collections, e.g., for long-standing (tracking) queries, and keeping track of a user's exposed location profile across time and space.

5.3 RFID Privacy Tools

When clothing manufacturer Benetton announced in March 2003 that it was considering the use of Radio Frequency Identification (RFID) chips in its garments in order to streamline its supply chain, an unexpected storm of protest followed in the media [71] that ultimately forced the company to withdraw its plans only a few weeks later [34, 99]. Similar statements followed in October of the same year from both retail giant Wal-Mart [62] and razor-blade manufacturer Gillette [64], after tests involving RFID-prototypes had been made public, which both companies had secretly been conducting in several retail stores.

[51] However, the latest move to allow for longer data retention for crime prevention and national security, as exemplified by the Directive 2002/58/EC [96], indicates that law enforcement agencies might want to retain more of such location data.

In all three cases, a previously unknown consumer protection group named CASPIAN ("Consumers against Supermarket Privacy Invasion and Numbering") had called for a world-wide boycott of the global companies. The fact that their campaign had such an immediate effect suggests the importance the topics of privacy and security have attained with the general public.

RFID tags represent a significant privacy problem – at least in principle – due to their enhanced means for identification. While proponents of this technology often like to compare RFID to the ubiquitous, yet by no means threatening bar codes, RFID does differ from them in two important respects:

1. *Level of Detail:* While special two-dimensional versions might carry up to 1000 bytes at the expense of larger print areas and lower reliability during scanning, the majority of today's barcodes feature only about a dozen digits. RFID tags in contrast store usually hundreds bits, and are already designed to carry not only a class-identification (e.g., a manufacturer-id and product-id) but rather item-level-identification (i.e., a serial number). Some types of RFID tags can even be rewritten.

2. *Unobtrusiveness*: Reading a barcode requires a line-of-sight between the reader and the tag. This means not only that the scanning process itself can hardly go unnoticed, but also that the tags must be easily visible. RFID-Labels in contrast are read from (or written to) through an electromagnetic field, which can easily penetrate plastic, fabrics, or paper. Thus, both the fact that a tag is present, as well as the act of reading out such an RFID-tag can be concealed.

Work on technical privacy-protection tools for RFID-tags has therefore focused on reducing the amount of detail reported by such tags, e.g., by replacing the stored serial number with a generic manufacturer code or even a completely arbitrary number, and on preventing any unnoticed read-outs of such tags. Due to the envisioned widespread usage of such tags, the former method might only be a partial solution: Even if the level of detail provided by such tags is significantly reduced, the specific combination of tags carried by an individual, so-called "constellations" [349], might still allow for the identification of a person.

Existing technical solutions in the field of RFID privacy can be divided into anonymizing and pseudonymizing methods. Both can either be achieved by deleting or altering the data on the tag itself, or by controlling read access to it. Especially the latter is critical, since RFID readers must also provide the energy to power the battery-less tags, resulting in reader-to-tag communication that stretches much further than the corresponding return channel from the tag back to the reader. The following sections introduce and evaluate the range of proposed RFID privacy solutions. Later, we will contrast these in chapter 6 with our own approach to RFID privacy – a so-called *RFID transparency protocol* – which follows the principles and mechanisms of our PawS prototype.

5.3.1 Kill-Command

Long before the Benetton incident triggered a public controversy over the use of RFID tags in consumer articles, the 2002 Auto-ID specification[52] contained the requirement of a "kill"-command [17]. The basic idea is simple: Before selling a tagged item to the consumer, the embedded tag is permanently deactivated at checkout. This renders the tag inaccessible to subsequent reader commands and thus prevents any tracking or profiling beyond the point of sale.

The current Auto-ID/EPCglobal specification[53] requires for all conformal tags an 8-bit-password to be set on the tag during or right after production in order to prevent unauthorized deactivation of the tags through this kill-command, e.g., while still on the shelves. After receiving the correct password, the specification requires the tag to stop responding to all subsequent reader commands in any way [18]. How this functionality is actually implemented on the tag is left up to the manufacturer, though due to cost efficiency, most solutions are currently software-based, which would allow – at least in principle – a later reactivation of the tag through direct contact (as the over-the-air interface is deactivated).

Apart from this potentially incomplete tag destruction, two additional aspects significantly affect the efficiency of this method from a

[52] The Auto-ID center was founded in 1999 to develop both RFID tags and standards for identifying everyday things, especially in the supply chain [122].

[53] Since the Auto-ID center's scheduled close in October 2003, the commercialization and further development of the Auto-ID technology is done by EPCglobal – a joint venture between the Uniform Code Council and EAN International [122].

privacy point of view. For one, deactivating the tags at checkout would still allow for detailed tracking of consumers inside stores, as well as associating consumer data and shopping information right at the point of sale (e.g., through the use of a credit or consumer card when paying). Additionally, the process of deactivation itself is for the consumer difficult to verify, as no visible cues would be present. The fact that all known deactivation methods are software-based, even though a permanent electro-magnetic deactivation similar to today's anti-theft labels would in principle be equally possible, is seen by critics as further proof that a later reactivation is left as a possibility – a suspicion that seems to have been already vindicated by some fielded prototypes: during a visit to Metro's Future-Store by RFID-activist Catherine Albrecht, a detailed inspection of the supposedly killed tag revealed that only Metro's own product number had been deleted, while the tag's hardware serial number was still left intact due to "technical reasons" [124].

Others point out that equipping all existing point of sales with "kill stations" is widely unrealistic [327], since small businesses such as kiosks would never be able to afford the corresponding equipment, even though they would inevitably sell tagged merchandise (e.g., soda cans or razor blades). Today's prototypes for tag deactivation are also not yet capable of handling multiple tags at once: not least due to the password-protection mechanism, customers must laboriously silence each individual tag manually – a nuisance that might prompt many customers to abstain from bothering with the deactivation.

Permanently deactivating tags of course also prevents any secondary use of such identifiers, e.g., as part of the often-cited intelligent fridge or other smart household appliances; for providing follow-up services such as automatically recommending matching accessories for tagged clothing; and to improve product life-cycle services such as repairs, returns, and recycling. A comprehensive use of RFID even after the point of sale would benefit not only manufacturers and retailers in the form of an increased consumption through countless smart fridges, but also consumers, who might appreciate being told of expiring produce, or to be able to simply return a defective product without having to worry about keeping the receipt (since the product's RFID tag stored all relevant data for the return).

While an impressively simple and seemingly effective method, the "kill-tag" approach has thus five significant drawbacks that might prevent the widespread adoption of this solution:

1. It is not easily verifiable by the customer, as the deactivation of the tag is not readily apparent.

2. Even if deactivated at the point of sale, substantial tracking concerns remain inside stores before a product is sold.

3. If implemented as an optional deactivation service, it puts a high demand on consumers, as these have to laboriously deactivate the items they bought.

4. A comprehensive deployment of "kill-station" seems unlikely, given the high cost of such stations for small-scale retailers.

5. Additional follow-up services after an item has been sold, such as improved returns or recycling, are rendered impossible once the tag is deactivated.

5.3.2 Hash Locks and MetaIDs

As an alternative to the "all or nothing" approach of the kill command, a number of proposals favor protecting the RFID-tag payload (i.e., the tag ID or, alternatively, the stored electronic product code) from unauthorized reading. As soon as a product changes into the hands of the consumer, a key-based access protocol would allow him or her to control who would be allowed to subsequently read out the stored tag information.

The basic principle was already proposed in 2002 by Sarma et al. [303] and is based on mathematical one-way-functions, so called "one-way hashes." A one-way hash takes an arbitrarily long input and computes an (often fixed-size) "fingerprint" or "digest" from it. While this computations is typically relatively easy to do, determining the original input given such a fingerprint is much harder, ideally it would be virtually impossible.[54] In order to "lock" an RFID-tag, an RFID-reader device would choose an arbitrary key k, compute a hash value $h = H(k)$ from it using a reasonably secure one-way hash function, and store this hash value (called the "MetaID") in a specially reserved area on the RFID-tag. In order to facilitate unlocking the tag at a later time, the owner (or better: his or her tag-reader) would also file the random key k under its MetaID h in a database under the owner's control. Once a tag has

[54] A well-known one-way hash algorithm is MD5, developed in 1994, which takes any input and computes a 128 bit fingerprint from it [292]. It is widely used to ensure data integrity for software distributions and email messages.

a stored MetaID, it replies to all read requests with only this MetaID, never with its "true" ID or any other data payload it might carry (e.g., its *Electronic Product Code* (EPC) – a standardized identifier that not only carries a manufacturer and product ID, but also a product's serial number [122]). If the tag owner later wants to access the original data again, he or she simply queries the tag for its MetaID h (which is the only information accessible from the tag, anyway) and looks up the corresponding key k that was originally chosen by the reader to lock the tag, using the database of key-MetaID pairs. Once this key k is sent to the tag in question, the tag will itself perform the computation of $H(k)$ and verify if it matches its stored MetaID. Should this be the case, it deletes the stored MetaID and is thus effectively unlocked again.

An access-control scheme using one-way hashes as keys has several advantages. Even though it does not offer absolute security in the mathematical sense, computing the original unlock value k from the stored hash value h requires such a substantial effort that for all practical purposes, being able to read out the MetaID h will not allow an unauthorized reader to deduce the original value k for unlocking the tag. Also, providing RFID-tags with the ability to compute a hash-value (for verifying that the reader-sent unlock value k does indeed form the basis for the stored MetaID $h = H(k)$) is relatively cheap to implement [349], and would thus also be an option for ultra-cheap RFID-tags – an important advantage over more complex (and therefore potentially more secure) solutions that use symmetrical or asymmetrical cryptography,[55] which are only an option for relatively expensive goods that can "afford" an expensive tag.

5.3.3 Variable MetaIDs

While MetaIDs effectively prevent unauthorized readers from accessing the "true" tag data (e.g., its EPC number), they nevertheless still allow the hidden tracking of tagged items and therefore potentially their owners. This is because even though MetaIDs block access to the "real" ID, their persistence makes it possible to repeatedly track an item as it passes several different readers.

An improvement of such static MetaIDs are so-called "randomized hash-locks" [350]. Their goal is to prevent the creation of detailed

[55] See for example the GenuID-tags from NTRU Cryptosystems: `www.ntru.com/products/genuid.htm`

tracking records by repeatedly accessing a fixed MetaID using several different reader devices. For this, tags do not reply with a fixed MetaID anymore, but instead generate their MetaID anew upon each read request from a tag-reader. An integrated random number generator on the tag generates a random value r_i, which is appended to the "real" ID of the tag and thus forms the basis for a temporary MetaID $h_i = H(ID||r_i)$. A reader receives both the temporary h_i as well as the used random number r_i. In order to deduce the real ID of the tag, the reader needs a list of all possible IDs – a requirement that seems feasible for individuals with a small number of tagged items (as opposed to large supermarkets with hundreds of thousands of tagged items in store). Using this list of known items, the reader device then simply computes $h_j = H(ID_j||r_i)$ for all its known IDs, until it finds an h_j that matches the h_i it read from the tag. With this, it implicitly knows the ID of the tag and does not even have to explicitly unlock the tag (which would work analogous to the fixed MetaID scheme). Only if an item would be returned or transferred to a different individual, the reader would send the found "true" tag-ID ID_j and thus unlock the tag again.

Even if the solution is not cryptographically robust, as an attacker could repeatedly read out the generated MetaIDs and infer the original ID from them,[56] it nevertheless fulfills two important requirements for RFID privacy: it prevents the unauthorized readout of the real tag ID and makes the tracking of tags (and with this of their owner) difficult at least.[57] In addition, integrating random-number generators in RFID-tags seems economically feasible, even for cheap tags, as the latest EPCglobal RFID-standard already requires tags to have one [18]. However, once the "real" ID of a tag is know (e.g., when returning it to the store), this method would allow the identification of the item (and with this also of its owner) through a retrospective analysis of logfiles.

[56] Weis et al. propose an extension to their original scheme, which would, in addition to the random number generator, carry a pseudo-random-function (PRF) ensemble f_k that would be initialized with a secret key k. Instead of directly computing the variable MetaID out of the ID and the random number r_i, the tag would XOR its ID with $f_k(r_i)$, i.e., the pseudo-random number that results if you seed the k-th pseudo-random-number function with r, before computing $h_i = H(ID_{XOR}||r_i)$. The reader in turn would need to know the secret k, which would allow it to select the correct PRF f_k, compute $f_k(r_i)$, and with this computes ID_{XOR} for all of its known IDs, before computing h_j as before. While this solution would be cryptographically robust, Weis et al. are skeptical whether PRF-ensembles could be cheaply integrated into mass-market RFID-tags.

[57] With a large number of tags present on a person, one could also imagine tracking the total number of tags, regardless of their MetaIDs.

As an alternative, Ohkubo et al. [257] propose so-called "Chained Hashes:" instead of repeatedly computing new MetaIDs from the same ID, tags would compute their new MetaID directly out of the previous one. In order to harden their system against attacks, they propose two separate hash-schemes: one for computing the new MetaID out of the current one (i.e., the new MetaID is computed as $ID_{i+1} = H_{chain}(ID_i)$), the other as an additional precaution to shield the computed ID_{i+1} (i.e., the reader receives $ID_{out} = H_{out}(ID_{i+1})$). By returning only a hash of the computed MetaID, an attacker should be unable to learn anything about the chain of MetaIDs that are computed internally.

As with the random-hash-lock method by Weis et al. above, a reader device would again need to know the IDs of all tags it wants to read out. In order to identify a tag in its vicinity, the reader needs to compare the received ID_{out} with its own list of MetaIDs, which of course would need to be hashed with H_{out} as well, i.e., the reader would need to repeatedly compute $ID_i = H^i_{chain}(ID)$ until $H_{out}(ID_i)$ would match the received MetaID. In order to speed things up, the reader could keep track of how often it has read a tag and store this together with the original tag ID. Alternatively, the tag could send the value of i along with the computed MetaID, though this might affect its robustness against attacks.

In contrast to the solution by Weis et al., where each answer is directly based on the "true" ID of the tag (which allows an attacker to reconstruct all existing log entries once the real ID is known), the chained-hash scheme uses the original tag ID only when generating the very first MetaID – all of its subsequently reported MetaIDs ID_{out} are hashes of hashes of this "true" ID. Should an attacker ever be able to read out the currently stored ID on the tag, he would still need to invert a (potentially large) number of H_{Chain} operations in order to find the true ID (and thus gain access to existing log entries).

A compromise between speed and security is favored by Henrici and Müller [163], who store the last two MetaIDs of a tag in a database and store a new random MetaID on the tag after each read. By using a *TransactionID* (TID) that is known to both the tag and the reader, and which is incremented by one after each read, they can both prevent replay attacks and encrypt the new MetaID when it is sent from the reader to the tag. In order to handle lost messages, the tag stores both the current TID, as well as the TID of its last known successful

transaction, and sends the difference between the two along with its (hashed) current TID as part of its answer to the reader device. The difference allows the reader to detect lost messages and thus prevent reader and tag to become out of sync: Should the newly computed MetaID (sent from the reader to the tag) fail to be set on the tag, the reader can detect this through a difference in the TID and reuse its last known MetaID to attempt another ID-rewrite on the tag. The advantage of this approach over Ohkubo et al. lies in its simpler tag hardware, as no special computation is done on the tag. However, its main drawbacks are the more costly data storage and synchronization requirements.

A much simpler alternative is proposed by Inoue and Yasuura [176], who opt for a completely random number chosen by the user (or, by one of the user's reader devices). Just as with the other hash-lock methods, setting such a private ID on the tag will lock the tag-contents from read requests. However, instead of locking and unlocking the tag over the radio channel (and thus having to implement a remote authentication method), Inoue and Yasuura propose to us a separate channel, such as direct contact or a very short distance (see Fishkin and Roy's work in section 5.3.4 below).

While this approach keeps the complexity of the tag minimal (as no hash function or random number generator needs to be implemented on the tag), their proposal increases the effort for the consumer slightly: even though ID management would most likely be handled by some user-controlled reader-system, the lack of authentication mechanism requires physical contact to each object whose tag should be rewritten. Especially if traceability is to minimized through repeated rewriting of the private ID, this contact-based authentication seems cumbersome. However, choosing a completely random ID instead of a hash-based one does make log information more robust.

A second variant proposed by Inoue and Yasuura minimizes the tag requirements even more by employing read-only tags. Instead of hiding tag-data with a private ID, their alternative mechanism uses physical tag separation: the unique EPC of a product is stored on two tags instead of one – one designating only the product class, the other tag containing only the serial number. Once these two are separated (e.g., if the product class in part of the package, while the unique ID is integrated into the product), the unique identification of a product that was still possible in the store is not possible anymore. However,

while this would prevent the popular "underwear-readout" example, the remaining serial number would still allow the tracking of items.

5.3.4 Access Control

A different approach to authenticating legitimate reader devices is put forward by Fishkin and Roy [118]: Based on the principle *distance implies distrust*, Fishkin and Roy propose tags that return more or less information based on the distance to the reader devices that poses the query. As an example, they list five possible levels of disclosure: At level zero, the tag only announces its presence. At level one, it replies with generic attributes (e.g., a shirt would reply with its color and fabric). Only at the highest level of four, personally identifiable information such as the location and time of purchase would be released.

For the actual distance measurement, the authors propose three different methods with varying advantages and disadvantages. The most reliable method seems to be triangulation, i.e., at least three time synchronized tags would relate their received signals to a base station, which would then compute the relative position of the reader[58] from the differences in the time-of-arrival for each signal, and send this information back to the tags.

The substantial infrastructure requirement for such a solution (a trusted base station in range, a cryptographic protection from illegal base stations, time synchronized tags, and at least simple signal analysis capabilities on the tags) seems to prohibit a realistic use of this approach in the foreseeable future, even if, as in their second alternative, the authors replace the comprehensive signal analysis with a simple signal-strength-measurement, which would be simpler, but also less reliable. Their third alternative would operate directly on the tags themselves, without a need for separate base stations: Measuring the standard-deviation of the signal-to-noise ratio (SNR), a tag could roughly estimate its distance from a reader devices, as the standard-deviation increases with distance. However, while this might seem the most elegant solution, it also entails the least reliable distance measurement: Even with both the tag and the reader device stationary, any dynamic environment would significantly affect the background noise (and thus influence the measured distance).

While the basic principle of their approach is rather simple, the prac-

[58] Relative to the tags, that is.

tical implementation is not. At the outset, the signal strength of a reader device at a tag depends heavily on the tag's orientation – as soon as it changes from its "optimal" position, the reader will appear much further away than it really is. While this might be tolerable from a privacy point of view (after all, more distance implies less data transfer), it would make reliable application design almost impossible.[59] Additional, both metallic substances and water[60] significantly influence the energy field of an antenna, which makes reliable measurements outside laboratory settings difficult. While the authors hope to increase reliability by combining the different approaches, and by putting more complex antennas on the tags, the difficult "user interface" of such a solution, as well as its increased cost, will most likely appeal neither to customers nor to service providers. That is because even with a reliable distance measurement, consumers would be unable to judge the actual information exchanged in everyday operations, where, in theory, leaning too close to a (potentially unknown) reader could accidentally disclose detailed information. This also prompts the question whether the hierarchical organization of tag-data is always useful or even possible.

5.3.5 Eavesdrop-resistant Anti-Collision Protocols

Due to the power asymmetry between reader and tag, information sent from reader devices would be subject to eavesdropping, even if using one of the above authorization methods, where only "friendly" reader-devices would get access to the information stored on the tags. This is because of the energy field of the reader, which not only transmits the information from the reader to the tags, but is also used to power them, and thus typically has a much larger range than the signal that is reflected back from the tag. This allows third parties to "listen in" on the signal sent from the reader, even from a considerable distance.

This is especially critical if the tag's ID is among the information sent from the reader to the tag. While this might sound unlikely at first (after all, it is the reader that is interested in the tag ID, not

[59] A good example are today's RFID-based, contactless ski passes: In order to prevent readers from picking up the pass of someone further down behind, the reading distances must be kept rather short. This inevitably forces skiers to rub their jackets containing their passes in a number of different positions against the reader until the RFID-tag is properly detected by the gate.

[60] As humans contain 45-60% of water, the presence of a only single user already "interferes" with the RFID-system.

the other way around), it is quite common practice in binary-tree-based anti-collision protocols [213]. As tags typically have no way of detecting the presence of other tags, their replies to a reader's signal might conflict with the signals from other tags in the vicinity, thus creating a "collision," an interference that prevents the reader from decoding the IDs of all of the involved tags.

A popular variant of such a protocol uses ID *prefixes* sent from the reader to determine which tags (i.e., only those with a common prefix) should reply. As long as the reader detects a collision (i.e., if two or more tags with the same prefix as indicated by the reader are within range), the reader increases the length of the prefix (e.g., by adding a "1" to it) until a single tag ID can be "singularized." It then replaces the bit it added last with its inverse and continues – should more collisions occur – to increase the length of the prefix. For example, should the tags "1001" and "1011" be in range, both would reply to an initial "zero-prefix" query by the reader, thus rendering their replies unreadable to the reader. The reader would then send the selection prefix "1", which would still have both tags reply (as they both begin with this prefix). Continuing with the prefix "11" would get no response at all, so the reader would try "10" instead, again resulting in a collision. Only when sending out "101" and after that "100," each of the two tags would reply individually. This explicit partitioning allows the individual selection of an arbitrary number of tags. However, the above asymmetric transmission power would allow a third party to log the sent-out prefixes, potentially learning the individual tag IDs should a collision occur at the very last bit position.

Weis et al. [350] propose that instead of sending a whole prefix, readers would only send the command "transmit next bit" to the tags. As long as their corresponding bit positions are identical, no collision would occur[61] and the reader would be able to note the common bit prefix incrementally. Once two tags would differ at position i, the reader would just as before use a "select" command to pick a subtree, but instead of sending the complete prefix to the tags (i.e., sending bits 1 through i, with either "1" or "0" at position i), it would simply XOR Bit_{i-1} with its chosen Bit_i and send the resulting value. Tags in turn would XOR the received bit with their own Bit_{i-1} (which must be identical to the reader's Bit_{i-1}) and compare the resulting value to their corresponding Bit_i. In case of a match, a tag would be selected and reply with its

[61] A collision only occurs if two tags send a different bit value.

Bit_{i+1}. An attacker who could only listen to the forward channel (i.e., who could "hear" the commands of the reader, but not the replies from the tags) would not be able to observe the bits of collision-free prefixes (since the reader only sends a "Send next Bit"-command and the replies from the tags are too weak to be detected over long distance). Similarly, such an attacker would be unable to deduce any bit-values in case of collisions, as the XOR with an unknown value (Bit_{i-1}) also hides the reader-selected subtree-bit at position i.[62] However, in order to "remember" the current bit position, tags would need to carry (expensive) dynamic memory.

An alternative anti-collision method can potentially work without sending out any information on the forward channel: In protocols based on the Aloha-Model, tags reply individually with a random delay to the reader signal [340]. Depending on the (reader-set) time allocated for tag-replies, tag transmissions distribute themselves randomly and can ideally be read collision-free. However, in order to increase the performance of such protocols, some variants explicitly "silence" tags that have been correctly identified, in order to lessen the number of tags that need to be read if only a few collisions occur. Unless special care is taken, such a selection mechanism would of course allow a distant attacker to log the IDs of such silenced tags.

The current EPCglobal tag specification [18] contains a requirement for a random-number generator on the tag, both for reasons of efficiency and security. Instead of its "true" ID (typically the EPC), the specification requires tags to reply with a random number that is generated for each read cycle anew. In order to "silence" a tag under this protocol version, the reader uses this random number. Once all tags have been identified using their momentarily chosen temporary IDs, readers can then use these numbers to request the "real" ID from each tag. This not only prevents attackers from "listening in," but also increases the speed of the anti-collision protocol as the temporary ID uses fewer bits (12) than the globally unique EPC (96) and thus provides for shorter transmission times.[63]

[62] As an example, consider the three tags 00101, 00001 and 00110. The only reader commands an attacker would hear would be: *GetNext*, *GetNext*, *GetNext* (Collision between Tag_1, Tag_3, and Tag_2), *Select(1)* (Collision between Tag_1 and Tag_3), *Select(0)* (Tag_1 identified), *Select(1)* (Tag_3 identified), *Select(0)*, *GetNext* (Tag_2 identified).

[63] This obviously only holds for large tag populations, as otherwise the overhead of reading out the EPC separately is too large.

5.3.6 The Blocker-Tag

Probably the simplest proposed access control method for RFID-tags is based on the above described binary-tree-based singularization protocol and follows a denial of service approach [186]. Juels and Pappu propose that consumers carry a so-called *blocker-tag* with them, which replies to any read request with a self-induced collision (using two antennas that reply with two conflicting IDs). Using the above mentioned binary-tree-based anti-collision protocols, readers would thus begin the task of singulating individual tags from the apparently large population of tags. However, for any prefix sent from the reader device, the blocker-tag would create a collision, therefore forcing the reader to traverse the entire tree of all possible ID combinations – when using a 96-bit EPC, it would have the size of several billions of tags. Even if a reader would be able to read several thousand tags per second, the presence of such a blocker-tag would effectively stall any read attempt indefinitely (or until the reader device would give up).[64]

In order to use this effect in practice, Juels and Pappu propose jamming only certain subtrees of the possible ID space, e.g., all tag IDs that begin with "1...". Instead of permanently deactivating tags at checkout (as proposed in the kill-tag approach), tags would simply have their first bit rewritten from "0..." to "1...", thus being sorted into the "private" space protected by the blocker-tag they are carrying.[65] Similar to the different information-zones proposed by Fishkin and Roy [118], this principle could actually be extended to create not only one, but a number of such privacy zones (using two or more bits for the prefix) that would be protected using different blocker-tags, or through a dynamically configurable super-blocker-tag.

In order to prevent readers from locking up when trying to read such protected subtrees, the authors propose a simple signalization scheme that could announce the presence of such a blocker-tag and the prefix it protects, e.g., using a reserved tag-ID that could be queried before the actual scan is started. Another problem is the possible interference of a blocker-tag with other people's tags, as anybody within its range would have his or her tags involuntarily blocked as well. Juels and Pappu propose using several dozens, if not hundreds of privacy zones

[64] Even an address space of only 64 bits would keep a reader capable of reading 100'000 tags per second busy for over four billion years.

[65] Being only slightly more expensive as an ordinary tag, supermarkets could already integrate blocker-tags into their complimentary paper bags.

(i.e., prefixes) in order to minimize the chance that two people carrying blocker-tags jamming the same subtree. However, increasing the number of distinct privacy zones increases the ability to track people not through their tags, but indirectly through their individual blocker-tags and their announced privacy zones.

The biggest advantage of the blocker-tag approach is certainly the minimal infrastructure that is needed: existing tags (at least those with rewritable memory) could be used unchanged, and reader devices would only need minimal software updates to cope with privacy zone announcements. On the other hand stands the rather poor reliability of such a method: by implementing blocker-tags cheaply as a passive RFID-tag, a slight misalignment could easily cut power to the blocker-tag and thus expose the formerly hidden tag population. Using cheaper, non-writable tags would keep costs further down, yet would greatly increase the interferences between blocker-tags and legitimate read operations: A neighbor helping with the shopping bags prevents my smart fridge to detect half of my groceries, and my smart laundry machine is unable to detect the proper program due to the blocker-tag I left in the pocket of my jeans. Equally possible seem advancements in reader technology that would allow readers to differentiate between "real" collisions and those that are simulated with a blocker-tag.

5.3.7 RFID Security

Besides the automated tracking capabilities of RFID-tagged goods, RFID tags are also used as an added security feature to thwart counterfeiting, e.g., in high-priced consumer goods such as designer clothing. Plans to incorporate RFID tags into Euro banknotes [381] and passports [377] have repeatedly prompted public concern, due to the sensitive nature of these items. Chips in banknotes are thought to make counterfeiting more difficult, but also help fighting money laundering [65]. In contrast to optical technologies, RFID chips are also thought to be more robust against wear and tear. Similar reasons are given for embedding RFID in passports, along with helping to fight terrorism [377]. Additionally, the contactless read capabilities of RFID chips offer longer lifetimes than the pins of a regular smart card [199].

RFID in Banknotes

Apart from recent confirmations about the type of chip that will be embedded in the Euro banknote (according to a Hitachi spokesperson, the European Central Bank (ECB) is planning to use Hitachi's μ-chip [153]), the only known fact is that the chips are supposed to carry a read-only "38-digit number" [153].[66] This renders mechanisms like hash-locks, MetaIDs, or kill-commands useless, as they require writable tags to deactivate or overwrite the original ID.[67] However, giving the (current) owner of a banknote control over the embedded chip would of course contradict the original idea of preventing counterfeiting. Even so, banknotes will probably pose less of a threat to privacy as this might suggest. Even without the help of a blocker tag, the exact number (and denominations) of banknotes an individual carried in her purse would hardly be detectable from a passing thief searching for the next victim. This is because the usage of RFID tags with large read ranges would actually be counterproductive for banks, merchants, and law enforcement agencies alike, as this would make it difficult to relate a digital ID that has been read with the specific banknote in hand. Not surprisingly, the chosen μ-chip has a read range of just one millimeter [288]. Even if tags with a slightly higher range were used, and thieves would use crowded subway-trains to approach their victims, a purse lined with aluminum foil would easily spoil such attempts. Even without such a protection, having several banknotes aligned and stacked would significantly detune each of the tags, thus thwarting any read attempt of the entire stack.[68]

[66] It is not yet clear what is actually stored on these tags. While 38 digits would be enough to store the 10-digit serial number, the (single-letter) country code of the issuing bank, the 6-digit "short code" (the short code identifies the printing origin, see `www.myeuro.info/euro-snr.php`), and any required checksum information, the complexity of synchronizing the printing process with the fab-initialized μ-chips might prompt the ECB to instead keep a database associating random chip serial numbers with banknote serial numbers after production [348].

[67] Notwithstanding, Jules and Pappur [185] earlier proposed a system using a combined optical and radio-based approach, which also requires writable RFID tags. The optical data consists of a printed access key, which is required in order to read and optionally write the information stored on the RFID chip. Without the key, only an encrypted serial number of the banknote can be read. Merchants are supposed to re-encrypt the serial number with a random number whenever they receive a banknote, in order to prevent tracking attacks. The random value is stored in the key-protected area of the banknote as well, thus allowing anybody with optical contact to the banknote to first decrypt the random value, and then decrypt the serial number (and, ultimately, to choose a new random value, re-encrypt the serial number, and store this new random value again). Avoine [19] has shown that the proposed mechanism does not actually require optical access to the banknote in order to successfully decrypt the serial number, and that attackers can still track such banknotes.

[68] This effect would also prevent any automated inventory taking of a whole stack of money in a bank, similar to the envisioned supply-chain stock-taking of RFID-tagged products, that

Another often-cited attack against RFID-enabled banknotes would be an increased, if not comprehensive, tracking of each individual banknote, including correlating each banknote to the person receiving or spending it. Merchants already have much easier tools at their disposal to learn individual shopping behavior, e.g., in the form of the increasingly ubiquitous loyalty cards. This is not only much cheaper than installing costly new banknote scanners, but also (and more importantly) legal, as consumers give their consent to such data collections upon signing the loyalty card application form. In order to execute such a scheme on a national, if not global scale, a central merchant-register for currency tracking would need to be installed – the number of parties involved in such a process makes this both economically and politically unlikely. The example given by Juels and Pappu [185] of several merchants secretly sharing their banknote data would not only be a severe violation of existing laws in many countries, but could again be implemented in a much cheaper and politically safer manner through a multi-merchant loyalty card, much like the card issued by the Payback group in Germany.[69] Similarly, fears of tracking banknotes through a writable "memory" chip that would "*allow money to carry its own history by recording information about where it has been, thus giving governments and law enforcement agencies a means to literally 'follow the money' in every transaction*" [16] seem unfounded, given the significant necessary investments in national and international monetary infrastructure to implement this, and of course the current chip's lack of writable memory.

RFID chips are thus only useful as another technical hurdle for reproducing counterfeit banknotes. Given the chosen, proprietary RFID technology from Hitachi, counterfeiters would need access to chip fabs capable of producing μ-chips with their 0.18 micron structures [348]. However, in order to detect a fake RFID chip (should counterfeiters ever be able to reproduce them),[70] or for following a "hot trail" of blacklisted money from a robbery or kidnapping, a central database run by the ECB might still be necessary, in which national and private banks, as well as merchants, might perform verification lookups.

some magazines alluded to [98].

[69] Payback loyalty cards are accepted at more than a dozen national retailers throughout Germany. See `www.payback.de`

[70] Once counterfeiters are able to incorporate an RFID chip with the right dimensions into a banknote, having it respond with the same (static) ID as a valid banknote is trivial to achieve, even if this ID has been cryptographically signed.

Such a central certification register would then be able to detect not only blacklisted IDs, but also identify duplicate banknotes if the same ID is submitted from two or more geographical places in too short a time that would allow for a single banknote to travel between these two places. Similarly, IDs that would be checked, on average, more often than others might also imply a duplicated banknote [195]. However, RFID tags in banknotes will probably not help the average citizen to better identify counterfeit money, as such chips would be embedded invisibly and thus only detectable with corresponding readers.[71]

RFID in Passports

In contrast to RFID in banknotes, embedding RFID chips in passports is already a reality. After the International Civil Aviation Organization (ICAO) approved the latest specification for "machine readable travel documents" (MRTD) in May 2004,[72] the US State Department began issuing RFID-enabled passports to diplomats and State Department employees from January 2005 [376]. On December 13, the European Union's Council of Ministers similarly decided to mandate that within 18 months, all passports issued in EU member countries must carry not only the MRTD-mandatory biometric facial image information, but also a digital representation of the holder's fingerprint[73] [268].

The EU plans also include another optional feature from the MRTD specification, namely an optical access control similar to the one proposed by Juels and Pappu [185] for banknotes: the access key for the RFID chip is computed from the already available machine-readable (through optical character recognition) data on the passport, the so-called "machine readable zone" (MRZ) [199]. Readers must first optically read the passport number, birthdate of the holder, and expiration date of the passport. After computing a hash value from this information, a reader contacts the RFID chip embedded in the passport to receive a random number, which it encrypts using the computed hash value. The reader also chooses a random number of its own, as well as one half of the session-key that should be used for the actual data transmission. Encrypting all three parts with the computed hash value,

[71] Though future mobile phones might include RFID readers capable of reading μ-chips and doing a lookup in realtime.

[72] See `www.icao.int/mrtd/`

[73] The MRTD specification requires that each passport carries a digital representation of the holder's facial image, and a digital signature from the issuing country. Countries can optionally also include fingerprints and iris scans [199].

the readers sends this back to the RFID chip, which in turn verifies that its own random number was correctly encrypted, after which it then decrypts the reader-chosen random number and the session-key part. The final step is then for the RFID chip to encrypt the reader-chosen random-number again using the hash-value, as well as a session-key part of its own, and send both back to the reader. The result is that both reader and RFID chip now have a complete session key (each half chosen by one of the two), upon which the actual data transmission can begin [199]. While the complexity of the hash-value used for decrypting this initial key exchange is high enough[74] to prevent an eavesdropping attacker from learning the chosen session key values and subsequently decrypting the actual biometric information, a recording of this communication could be attacked with more time and increased computing resources, in order to first deduce the initial hash value, and with this the session keys used for the actual data transfer [199].

Another complication arises from RFID-enabled visas, which, according to EU plans, should use similar mechanisms to increase their authenticity [220]. However, just as several stacked RFID-enabled banknotes will detune the individual tags so that reading all tags becomes almost impossible, the combination of an RFID-enabled passport with one or more RFID-enabled visa stickers will make the automatic reading process highly unreliable [219].

In contrast to RFID chips on milk cartons or clothing tags, the application of contactless identification technology in passports could have significant security implications. While the use of an optical key will most likely prevent *"that pickpockets, kidnappers and terrorists can easily – and surreptitiously – pick Americans or nationals of other participating countries out of a crowd"* [310], a determined attacker might still learn the data required to compute the optical key (passport number, birthdate, passport expiration date) for a particular individual and, using a sufficiently powerful reader, quickly scan a group of people.[75]

[74] Kügler [199] compares the complexity of the MRZ-based information to a 56-bit key such as DES.

[75] Again, using a face recognition system capable of identifying individuals in spite of superficial changes in appearance (such as mustaches or hair color) might be more reliable, as it also does not require the target to carry his or her passport with her.

5.3.8 Summary

RFID is probably one of the most prominent ubiquitous computing technologies today, owing to its widespread use (or planned use) in industry and its direct effect on consumers. The traditional, security-only based privacy solutions presented in this section often fail to be practically viable: Fishkin and Roy's distance-based authentication principle [118] seems appealing due to its intuitive simplicity ("distance implies distrust"), though it is most likely infeasible to realize technically, let alone reliably controllable for the consumer. Blocker-tags [186] are equally unreliable, as a slight misalignment of the blocker tag can quickly reveal the entire protected tag population.

More reliable and robust are the proposed hash-lock and MetaID mechanisms [303, 349], which make involuntary data disclosures unlikely as the "real" ID of an item is never revealed. However, MetaID solutions require not only a more complicated infrastructure setup, but are also not able to prevent tracking attacks using "constellations" of tags. Variable MetaIDs [163, 176, 257, 350] remedy this by providing a different number on every read, yet greatly increase overall system complexity, as all ID changes need to be tracked in a database. Also, users will need to engage in detailed tag management in order to properly register or unlock tags for the various applications they are allowed to work in (e.g., groceries stored in a smart fridge, clothes washed in a public laundry, or goods returned to a department store for exchange).

While the general idea of the kill-feature at first looks much simpler, it also requires a substantial management overhead due to its password-protection requirement (i.e., preventing unwanted silencing of tags, e.g., in a supermarket) that will most likely be impractical in many situations. A manual removal of the tag, e.g., by placing it on a removable label, is much simpler to implement and substantially more user-friendly, as it does not require specialized hardware and can be visually verified. This, however, prohibits value-added services after checkout.

While we also described the usage of RFID tags in security-related domains such as banknotes and passports above, the focus of our work is not on security applications. For these domains, strong cryptographic protection mechanisms are and will continue to be necessary, making some of the discussed RFID-privacy mechanisms a useful alternative, even though their use for groceries or clothing would be too costly.

5.4 Summary

This section has shown that a variety of options exist for providing technical privacy protection. However, only a few systems explicitly try to provide a comprehensive privacy infrastructure as we have proposed with PawS.

While Hong and Landay's Confab Toolkit [167] provides a comprehensive programming support for building context-aware applications, the proposed mechanisms are geared more towards peer-to-peer data sharing than institutional and commercial services. Similar to CMU's Aura project [161], the privacy preferences in Confab do not take established information disclosure practices into account, but only provide recipient, location, and time of day rules (e.g., "Only show Bob my location when I'm in the office on weekdays between 7am and 4pm."). Myles et al.'s work [249] follows our system closely, though with a more explicit focus on location data.

Identity management systems such as the *Freiburg Identity Manager* [181] or Lederer's *Faces* metaphor [214] offer valuable approaches on how a user interface for a privacy assistant in PawS might be constructed. Most work in privacy interfaces has so far focused on dedicated communication systems, such as Interliving's mirrorSpace [301], Neustaedter and Greenberg's Home Media Spaces [252], or Belotti and Selen's RAVE System [32]. PawS addresses a more service oriented application space, though it often follows the same general concepts of feedback and control implied by the Fair Information Practices (see section 3.2.2).

The concept of privacy-aware databases such as PawDB has received increased attention since 2002. Agrawal et al.'s *Hippocratic Databases* [9] and IBM's *Enterprise P3P* [192] follow a concept very similar to PawDB, using P3P-metadata to govern data processing in corporate databases. However, work by Sweeney [331] demonstrates the need for robust anonymization techniques in order to prevent weakly anonymized data from being merged later.

While alternative authentication concepts based on human trust instead of passwords or digital certificates have often been proposed (e.g., [103, 137, 142, 316, 317]) the complexity of interpersonal, real-world trust (see our discussion in section 3.1.2) has so far been too vague to implement effectively.

With RFID systems and location systems, two prominent examples of

ubiquitous computing technology received increased public attention, especially with respect to their privacy implications. Our concepts and mechanisms developed in chapter 3 and 4 seem well suited for these types of applications.

Location privacy solutions often aim at offering the user novel mechanisms to automatically control the dissemination of his or her location information, similar in scope to the overall concept of PawS. However, as the work by Sweeney [331] and Beresford and Stajano [36] shows, completely preventing the corellation of such information is quite hard, even if pseudonyms are randomly generated and often changed. Even "obvious" solutions such as self-positioning systems are no panacea, as service usage ultimately makes disclosing such self-collected information necessary, thus making it no more a complete "solution" than encryption and passwords are for informational privacy.

Existing RFID privacy concepts such as the kill-command [17], hash locks [303, 349], or blocker tags [186] all shift the burden onto the consumer, who needs to laboriously deactivate tags, reprogram them, or hide them with the help of an (unreliable) blocker tag. While strong cryptographic mechanisms are an important part of RFID systems in domains such as banknotes (to prevent counterfeiting) or passports (to prevent identity theft), typical supermarket scenarios will most likely not benefit from "secure" but impractical solutions.

What should become apparent from the range of related work described in this chapter is that no single mechanism is able to offer a fool-proof solution. However, combining a feedback and control tool such as PawS with an intuitive user interface, a robust pseudonymization mechanism, and a reliable privacy-aware database might be a good starting point for a privacy solution that is supported by (and provides support for) an effective legal privacy regime anchored in our moral ethics and norms.

In the next chapter, we will thus explore how our PawS approach might alternatively be used in the domain of RFID privacy. By using legal mechanisms to force reader operators to provide declared privacy policies as part of every read request, and by offering technical mechanisms to limit the amount of information readers request, we might not be able to protect against unauthorized read attempts, but will make it possible to detect unlawful reads and allow interested parties to obtain detailed logs of their daily invisible RFID interactions.

6 Applying PawS to RFID Privacy[1]

With the coming of a wired, global society,
the concept of openness has never been more important.
It's the linchpin that will make the new world work.
Peter Schwartz and Peter Leyden[2]

We designed an alternative privacy scheme for RFID systems, based on our PawS framework and along the principles developed in chapter 3. Instead of the all-or-nothing tradeoff of a kill-command, we use our basic concepts of a *privacy beacon* and a mobile *privacy assistant* to provide feedback and control to data subjects in RFID environments. Instead of deploying a special purpose beacon device, however, we opt for the *service protocol* alternative discussed in section 4.4 above. In the case of RFID systems, we can incorporate our data collection announcements directly into the reader-to-tag protocol, thus providing three core benefits:

1. RFID-system operators will be able to deploy readers that only collect tag data relevant to the actual application.

2. Data subjects can use mobile personal devices to receive detailed information about a reader's operator and its purpose for collecting data.

3. Future tags might be able to independently decide whether or not to reply to a reader's query, based on its stated ID, purpose, and target range.

Having RFID readers explicitly declare the scope and purpose of the tag data collection, as well as disclosing the identity of their operators,

[1] This section is based on joint work with Christian Flörkemeier and Roland Schneider [123].
[2] In [313]

Protocol extension	Init round all	SUID flag	Round size	CRC-5
1 bit	6 bits	1 bit	3 bits	5 bits

Figure 6.1: *The inventory command*, `Init_round_all`, as specified in ISO 18000-6 Type A. The command frame indicates the number of time slots that are available for a reply (round size), sets various flags, and contains a cyclic redundancy check (CRC) to detect transmission errors [123].

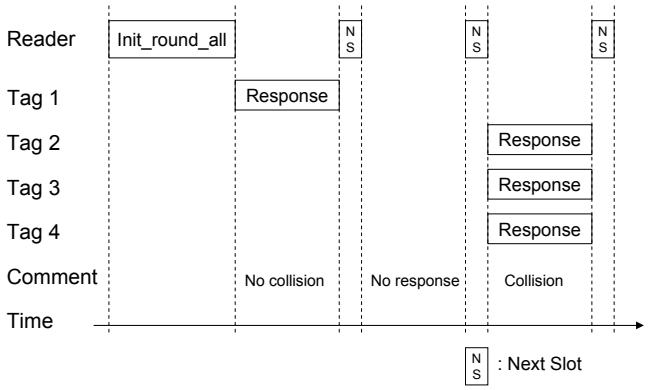


Figure 6.2: *The inventory process*, as specified in ISO 18000-6 Type A. The reader initiates a round of tag replies by issuing an `Init_round_all` command. Energized tags respond by selecting one of the available time slots at random to transmit their ID [123].

will allow both consumers and regulators to better assess and control the impact of everyday RFID encounters. The following sections briefly summarize the relevant protocol characteristics of RFID reader-to-tag interfaces, outline how we incorporate privacy policies into such a protocol, and describe how a mobile user device called a "Watchdog Tag" could be used to read out such information and provide corresponding feedback to the user.

6.1 RFID Protocol Primer

Once an RFID tag is within the read range of an RFID reader, the tag is powered and is ready to communicate with the reader. When multiple tags respond simultaneously to a request from the reader, their signals can interfere with each other, resulting in a failed transmission. In order to inventory all tags within the read range, an anti-collision algorithm that controls access to the shared radio channel is employed by the reader.

Figures 6.1 and 6.2 show examples of the inventory command (`Init_round_all`) and inventory process, respectively, as defined in the ISO-Standard 18000 Part 6 Type A [179] (which is the standard we are basing our protocol extension on). This standard uses a probabilistic anti-collision protocol scheme, meaning that tags respond at randomly generated times, e.g., based on the Aloha scheme [117]. Deterministic algorithms, in contrast, typically use a binary tree-walking scheme to traverse the set of all possible tag numbers (see section 5.3.5 above).

6.2 An RFID Transparency Protocol

We will use the ISO 18000 Part 6 Type A protocol as an example, and extend it with four concepts to support a number of Fair Information Principles:

1. An *identification* concept to support the principle of openness and accountability;
2. A *purpose element* to support the principle of a purpose specification;
3. A *collection type* that offers better use limitation; and
4. A *tag selection* mechanism to ensure collection limitation.

6.2.1 Openness Through Reader-Policy-IDs

None of today's RFID standards allows tags to identify the reader they are communicating with. The anonymous broadcast by the reader is certainly desirable from a performance point of view, since the reader's goal is to identify as many tags as possible in a certain period of time. The transmission of any additional data such as the identification number of the reader will thus reduce the speed at which tags can be detected. Without knowledge about the device that is collecting data, it is, however, impossible to satisfy the Fair Information Principles of *openness* and *accountability* (see section 3.2.2). In order to address these Fair Information Principle requirements also at RFID reader-to-tag protocol, we include a unique reader policy ID (RPID) into the reader's inventory command, which not only describes the policy in place (i.e., the *privacy contract ID*), but also uniquely identifies the reader and its operator. Since we will not be able to include a full

XML policy into the protocol for performance reasons, having this explicit reference to the policy allows us to provide additional information over a separate channel (e.g., WLAN). Also, the explicit reader ID facilitates dispute resolution by allowing customers to directly identify not only the policy used, but also the reader performing the request.

The RPID itself is encoded in a three-tier format, specifying the following three fields: the data collector ID, the policy ID, and the reader ID (cf. figure 6.3). With this structure, our solution follows closely the well-established EPC format and its general identifier encoding (GID-96) [105]. Even though we are not identifying products, but data collectors and their policies, this symmetry could potentially benefit the administration of the data collector IDs, as their identical format would allow data collectors to reuse their existing "General Manager Number" [105] of their EPCs (data collectors that do not already have such a number could acquire it in a similar fashion as they would for obtaining an EPC identifier). Moreover, EPCglobal's existing ONS architecture [239] that provides a look-up functionality for captured EPCs could transparently be used to resolve our reader policy references as well.

The policy ID follows directly after the data collector ID, giving data collectors a 24 bit value for identifying policies. Data collectors are free to substructure this value in any way they like, as they can do for the last value, the actual reader device ID, which comprises 36 bits. Useful substructures would be a division across country, region, city, or store, thus simplifying both policy publishing and reader localization from this ID. In our prototype, we use the policy ID to acquire more detailed policy information over wireless LAN, while the reader ID is resolved to its designated approximate location, in order to allow the (manual) detection of reader ID spoofs (e.g., a reader of a retail outlet on 5th Ave. suddenly appearing ten blocks south of this address).

Figure 6.3 shows a summary of our reader and policy identification code, and illustrates its usage again using the inventory command of the ISO 18000 Part 6 Type A protocol as an example.

6.2.2 RFID Purpose Specification

The FIP require that the purpose for which personal data is collected should be specified no later than at the time of data collection. P3P addresses this issue by providing a list of 12 abstract purpose types

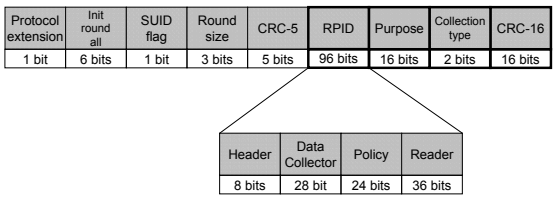


Figure 6.3: *The modified inventory command*, `Init_round_all`, of ISO 18000-6 Type A featuring an additional field for the reader policy identifier, the purpose declaration, collection type, and an additional checksum (CRC) [123].

that describe why data is being collected relevant to the specific web site that the policy describes (cf. section 3.3.3). Although RFID needs to be treated slightly different in the sense that in most cases the user will be unaware of the data collection taking place, as well as of the actual data being collected, many of the P3P purpose definitions can be equally well applied to the RFID domain.

Contrary to Web services, however, some purposes such as "admin" or "current" are much more difficult to assess in an RFID environment. For example, the current purpose is usually implicitly defined by the Web interaction the user is currently experiencing, e.g., the shopping cart checkout in a Web shop, while administration is usually defined by keeping Web server log files. In an RFID context, however, many different "current" or "admin" purposes can be envisioned: A smart shelf might issue read commands for inventory purposes (in a supermarket) or for asset tracking (e.g., for multimedia equipment that employees can check out from a central magazine), both of which could be called administrative purposes. "Current" purposes can equally vary, from a payment purpose at a self check-out station, to a repair and return purpose at a customer information station.

Consequently, we have expanded some of the existing P3P purposes while dropping others, in order to better reflect the more implicit interactions present in RFID systems. Table 6.1 lists the 14 purposes we identified as useful declarations in this context, even though additional purposes might become necessary in the future. This list is therefore only an initial suggestion that should be repeatedly validated by real-world prototypes, and subsequently standardized by an appropriate standardization body.

Apart from the "profiling" purpose, all purposes are encoded as single bit values that can be arbitrarily combined in our 16 bit number,

indicating that data are collected for multiple purposes. The profiling purpose uses three bits to encode one of five possible profiling purpose types that are mutually exclusive (see table 6.2).[3]

For example, a smart shelf application that monitors its contents for out-of-stock warnings, as well as provide data for anonymous in-store movement information (e.g., to see where consumers spend most of their time), would need to declare both the "inventory" and the "pseudo-analysis"-profiling purposes. A corresponding smart shopping cart that would provide customers with shopping suggestions, based on its contents, would declare "pseudo-decision"-profiling. And a self-checkout station that allows customers to wirelessly pay for their goods, while also associating the purchased items with the customer's loyalty card, would consequently declare the "payment," "anti-theft," and "individual-decision"-profiling purposes.

6.2.3 Use Limitation Through Collection Types

The principle of RFID reader-to-tag interactions (i.e., readers issuing an inventory command and tags replying with their IDs) makes it difficult to create privacy-friendly monitoring applications even if no identifying tag information needs to be collected as part of the envisioned application. Imagine an RFID system that tries to keep track of the number of people on a certain station platform, in order to avoid overcrowding. Even though RFID tags entering and exiting the area might reply to reader commands with their IDs, the application only needs to keep track of individual tags (e.g., an RFID-based train pass) without having to actually know their specific ID. Additionally, even when identifying information is collected, consumers will typically become much more concerned if this information is not only used locally, but also correlated across multiple readers in order to track an item's (or a person's) movements over time.

To allow data collectors to differentiate between the various collection needs, i.e., whether or not they actually require the serial number of individual tags, or whether they intend to track multiple occurrences of the same tag across different location, we additionally define four distinct collection practices that must be declared as part of a reader's inventory command:

[3] All extended purposes are of course also added to our privacy contract XML schema in order to allow privacy assistants to download the full XML version of the policy.

Type (Pos)	Description
access control (0)	Tag IDs are scanned for the purpose of access control, e.g., by identifying a pass holder or by authorizing the validity of an access key.
anti-counterfitting (1)	Readers read out data stored on the tags to assert the genuineness of a merchandise.
anti-theft (2)	Readers scan for tags that are attached to items that have not been paid for.
asset management (3)	Contrary to inventory purposes, tags are read to provide a picture of the whereabouts of assets, instead of monitoring changing stock quantities.
contact* (4)	Tag contents are read out in order to determine a contact channel to the customer, e.g., a mobile phone number or email address.
current* (5)	Tags are read to provide a service that was explicitly desired by the individual, e.g., when placing shopping items on a kiosk in order to calculate totals, or for disabling (killing) tags.
development* (6)	This purpose should be used during system testing and development only.
emergency services (7)	The system is monitoring tags in order to provide rescue workers with occupancy information.
inventory (8)	A shelf monitoring its contents, e.g., in order to provide out-of-stock notices to a central system.
legal* (9)	Law enforcement or other legal obligations require the system owner to read out tag IDs. Additional information on the legal grounds should be made available to the customer.
payment* (10)	The current action involves payment, e.g., at checkout when tag IDs are read for billing purposes.
profiling* (11-13)	Data is collected for profiling or ad-hoc personalization. See table 6.2 for individual values.
repairs and returns* (14)	Warranty and manufacturing details are read out in order to facilitate or speed up a repair or return process.
other* (15)	None of the above purposes fits. Further information should be accessible, e.g., in form of a sign or explicit contractual agreement.

Table 6.1: *RFID purpose declarations.* Data collectors can combine 15 different purpose declarations for RFID reader queries. Marked purposes have been taken directly from the P3P specification. Note that the P3P purposes `admin`, `historical`, and `telemarketing` have been left off this list as they only marginally apply to RFID systems. See the P3P specification [79] for further details [123].

Type (Bits)	Description
ad-hoc-tailoring (011)	This applies to immediate and anonymous tailoring, e.g., providing shopping recommendations based on the current content of a shopping basket, or suggesting accessories based on the clothing the customer has taken into the dressing rooms.
pseudo-analysis (100)	The collected data are used to learn about the interests or other characteristics of individuals. This may help to reveal the interests of visitors to different areas of a store. For example a store's shelves could be newly arranged based on the collected aggregated data.
pseudo-decision (101)	This information will be used to make customization decisions based on the interests of individuals, without actually identifying them. For example, a shop could suggest items to a customer based on his or her previous visits (without actually identifying that person).
individual-analysis (110)	The data collected is used in combination with identified data of an individual, allowing a profile of a certain customer to be generated. This could help to reveal the interests of visitors based on their age, social situation, or other relevant demographic data. Identification could occur in combination with a consumer or credit card.
individual-decision (111)	The information is used to determine individual preferences and to link them with identified data. This profile allows personalized suggestions, based on the individual's interests collected from previous visits, combined with personal information, e.g., from a consumer loyalty card.

Table 6.2: *Profiling purposes.* Profiling purposes are mutually exclusive, as profiling types lower in the table (i.e., with higher bit-codes) can potentially include all of the above types [123].

1. *Anonymous Monitoring:* Collecting state information about the items in the vicinity of a particular location, without the need to actual identify tags by their unique serial number. Examples would be simple sensor applications (e.g., an automatic door opener) or counting tasks (e.g., monitoring the number of items in a certain area).

2. *Local Identification:* Tag IDs are collected in order to provide a localized service, e.g., a smart medicine cabinet or smart fridge that monitors its contents. Although unique IDs are collected (e.g., for resolving them to human readable descriptions), the application does not require (nor attempt) the correlation of events across different locations.[4]

3. *Item Tracking:* Collecting information about the location of an item for the purpose of monitoring its movements. Note that this potentially enables tracking people through constellations. However, in order to differentiate between these different intentions, the separate "tracking person" declaration should be used, if people are tracked by the items they carry.

4. *Person Tracking:* Collecting information about the location of a person. Note that although item-level tracking can potentially imply the tracking of a person, data collectors would only need to declare this purpose if they actually collected RFID tag information with this application in mind. It is up to legal frameworks to force data collectors to anonymize item-tracking data so that it cannot be used for person tracking.

Together with a corresponding purpose, collection declarations further facilitate the accurate assessment of an RFID scan event. This does not only help data subjects to better understand the *intentions* behind a data collection, but can also be used to selectively allow tags to remain *anonymous* whenever possible. Anonymous replies are already part of some RFID protocols, e.g., ISO 18000 Part 6 Type A, though the reason for using them is usually efficiency, not data privacy. To detect collisions, a globally unique ID is usually not needed and just decreases the number of individual tags that can be successfully detected per unit of time. The anti-collision routine can thus first use the

[4]Note that if it is possible to combine logfiles of several location, item tracking would be possible and would thus need to be declared (either as "item tracking" or "person tracking").

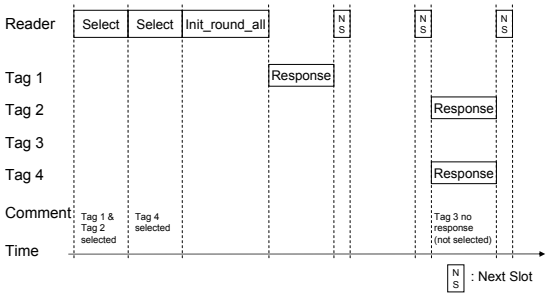


Figure 6.4: *The modified inventory process.* The reader first selects a tag population, before initiating a round of tag replies by issuing the modified `Init_round_all` command. Previously selected tags (tag 1, 2 and 4) respond in a randomly chosen slot [123].

tag's random short identifier to single it out from the set of present tags, before requesting additional data, which might include the unique, but static serial number. This kind of an anti-collision protocol could instead become the default whenever "anonymous monitoring" intentions are declared, thus explicitly providing tag anonymity and unlinkability.

Even without any specific support in the tags themselves, declaring "local identification" would still provide the data subject with the additional level of assurance that her movements would not be tracked across different locations (though this might not preclude the keeping of log files that could be later combined, e.g., as part of a criminal investigation). Obviously, none of these declarations are a proof that the data collector stating them is actually following them. However, as with the purpose declarations, any explicit privacy policy declaration provides a lever to threaten wrongdoers with legal actions – just as it is the case with today's printed policies.

Keeping with the examples from the previous section, the smart shelf tracking inventory and performing anonymous movement analysis of customers within the store would thus need to declare a collection practice of "person tracking", even though these traces are anonymous (pseudo-analysis). The smart shopping cart would use "local identification", as it would use the identity of the items in the cart to locally decide what other products to suggest to the user. Note that it does not matter whether this decision process is actually done on the shopping cart itself or wirelessly via a remote system, as long as the tracked tags are not correlated to other carts or shelves. A smart check-out station would need to declare "person tracking" again, in case a consumer loyalty card is scanned at the point of sale.

6.2.4 Collection Limitation Through Tag Selection

The first of the Fair Information Principles requires data collectors to limit the amount of data they collect to what is absolutely necessary (the EU directive makes this a legal requirement in most European countries, cf. section 3.2.1 above). Consequently, rather than asking *any* tag present to respond to a reader query and then filtering out the tags of interest on the application level, we want readers to limit their initial query to target only relevant tags in the first place, thus realizing the collection limitation principle already at the protocol level.

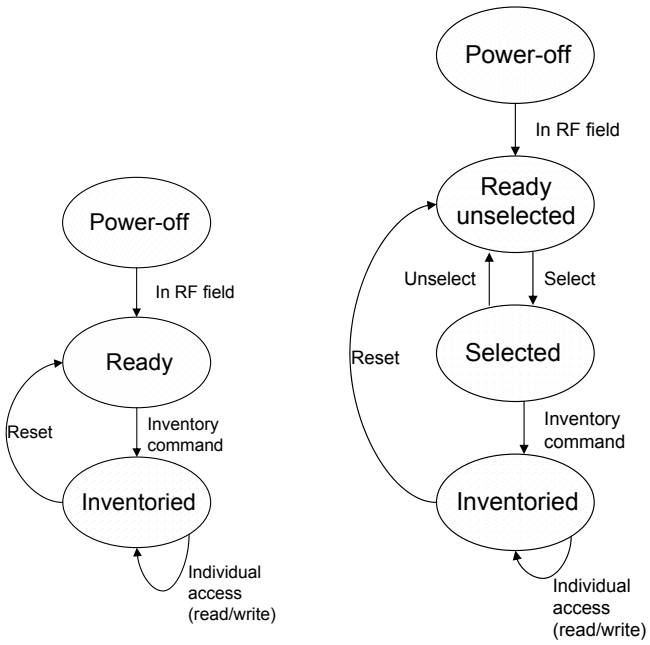
As an example of how this would work in practice, we use the frequently considered usage scenario of a supermarket smart shelf, whose purpose is to detect whether it is stocked with sufficient supplies of a particular item. Instead of issuing indiscriminate read commands, which might also pick up tags in the clothing of nearby shoppers, the shelf reader will target only tags of products stacked on the shelf, such as a particular brand of razor blades. Optionally, the shelf reader could occasionally run a separate request that targets *all* of the supermarket's products in order to detect misplaced items.

To implement this functionality in our reader-to-tag-protocol, we make use of a similar mechanism that is typically used to singularize a particular tag from a set of tags in range (e.g., the Group-Select and Group-Unselect commands in ISO 18000 Part 6 Type B). However, instead of using a selection mask to facilitate and potentially speed up the inventory process, we are using selection masks to restrict tag ID collection by the reader to relevant tags for privacy reasons.

Once tags appear in the range of a reader and get energized, they initially begin in an "unselected" state. Unselected tags will need to be explicitly selected before replying to any inventory, read or write command from the reader. Tags become selected only after receiving a select mask that matches their data in memory. Readers thus begin any command cycle with one or more select commands that first determine the tag population that is the target of the query (see figure 6.4). Once selected tags have been "inventoried", readers can issue actual access commands (see figure 6.5).

The `Select` command contains the following parameters (as shown in figure 6.6):

- *Pointer, length, and mask (PLM).* Pointer and length address a certain tag memory range. The mask, which must be "length" bits



(a) Typical (b) Modified

Figure 6.5: *Tag state transition diagram.* As soon as tags enter the reader's RF field, they move into the *ready* state and reply to the reader's "inventory" command, as shown in subfigure a). In our modified protocol, tag first enter the *ready unselected* state after getting energized, as shown in subfigure b). The tag moves into the *selected* state if it receives a matching "select" command. Only selected tags will respond to an "inventory" command by the reader [123].

long, contains a bit string that the tag must compare against the contents of the specified memory location.

- *Selection type.* The selection type indicates whether tags that match the PLM should enter the selected state or return to the ready, but unselected state.

An appropriate selection of tags that fulfills the requirement of the collection limitation principle will only be feasible if the tag IDs follow a known structure that allows for a certain grouping, e.g., a common prefix for a certain product from a particular manufacturer. This is the case in the currently favoured EPC system, where ID ranges are grouped by manufacturer ID and product type. If there is no such information encoded in the identifier, it needs to be available in the remaining portion of the tag memory and accessible during the selection process, as random tag IDs would be difficult to select efficiently.

Protocol extension	Select	State flag	Pointer	Mask length	Mask value	CRC16
1 bit	6 bit	1 bit	8 bits	8 bits	variable	16 bits

Figure 6.6: *The new* `Select` *command* enables readers to select a subset of tags within the read range. The state flag indicates whether a tag with a matching mask should enter or leave the selected state [123].

6.3 An RFID Privacy Assistant

In order to make full use of the additional information now present in the reader protocol, we use our concept of a *privacy assistant* in the form of a so-called "watchdog tag" to provide transparency to the otherwise invisible tag detection process. Simply speaking, the watchdog tag is a sophisticated version of an ordinary tag, as it features an additional battery, a small screen, and potentially even a long-range communication channel. The watchdog tag's main task is to decode the commands transmitted by a reader, and make them available on the screen of the device for inspection by the user (as shown in figure 6.7), or to log all data transfers and provide consumers with detailed summaries whenever needed. While the watchdog tag could be carried by the user as a separate device, its functionality could also be integrated into a mobile phone, allowing it to leverage the existing display, battery, memory capacity and long-range communication features of the phone.

Without the privacy features in the protocol, the watchdog tag would only be able to inform the user that some anonymous reader is scanning for tags in a certain vicinity. Due to the privacy features introduced in the RFID protocol, this notice can now include the operator's ID, the purpose and type of data collection, and the target range of tags. If a separate long range communication channel is available (e.g., wireless LAN or GSM), the watchdog tag can additionally translate the data transmitted over the RFID channel into a more expressive format, as shown in figure 6.7, simply by contacting an ID resolution service that translates the RPID into a service proxy URI.[5] In addition, providing the reader location in a human readable format allows for a simple, manual detection of reader ID spoofs. More sophisticated watchdog tags featuring an integrated location system could potentially detect reader ID spoofing automatically.

The above screen shots were taken from our initial watchdog proto-

[5] One such infrastructure would be the ONS architecture developed by the Auto-ID Center [239].

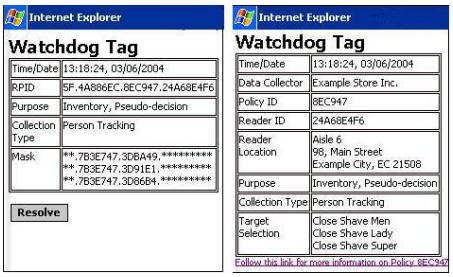


Figure 6.7: *The Watchdog Tag* The screen shot on the left shows data collected by the watchdog tag over the RFID channel. If a separate communication channel is available these raw data can be resolved to a more expressive, human readable format as shown in the screen shot on the right [123].

type,[6] which serves as our design test bed for our protocol extension. Built on top of a standard WindowsCE-PDA, it uses the built-in wireless LAN to retrieve human readable descriptions from a pre-assigned URI representing the *service-proxy*.[7]

6.4 Feasibility and Future Work

Our proposed protocol extensions are easily realized even with today's readers, as they only require updates to the reader's firmware, since the physical layer remains unaltered. While tags would require changes to their logic, these should be straightforward to implement, as the physical layer is not affected and only slight alterations to the medium access layer and the command set would be necessary. Our extensions do, however, affect the performance of an RFID system. The addition of the RPID, purpose code, and collection type require the additional transmission of 130 bits. At a data transfer rate of 30 kBit/s, typical for reader-to-tag signalling of systems operating in the UHF band, it prolongs the execution time of any command by 4.3 ms. This delay is thus comparable to the time it takes for a single tag to reply with its ID, assuming symmetrical data transfer rates. In modern RFID systems that typically read several dozens, if not hundreds of tags at a time, loosing a single tag slot thus seems negligible. For an RFID system that features a slow data transfer rate, e.g., 1.6 kBit/s as specified in ISO 15693 (HF), the delay is more significant, approximately 80 ms.

[6]The RFID watchdog tag prototype was developed as part of the diploma thesis of Roland Schneider [308].

[7]While subsequent student projects are exploring the use of a separate antenna design that would allow us to interface our PDA directly with the RFID reader's communication channel, the current system simulates the complete RFID protocol over the wireless LAN as well (with a PC posing as a virtual RFID reader).

However, in many situations such a delay would be outweighted by the shortened reply times, as the `Select` command allows the reader to ignore tag IDs that are of no interest to the application in the first place. Newly arriving tags in the read range will have to wait for the next select command before they can be inventoried by a reader.

Future tags might also be able to incorporate basic cryptographic functionalities, thus facilitating a national or even supra-national (e.g., EU-wide) certification system for IDs, as well as allowing tags to thwart an imposter's attempt to "steal" the identification string of a valid reader (thus supporting the Fair Information Principle *security*). To this end, companies would need to register their identification strings with the corresponding authorities, which would use their private keys to sign the submitted ID. Tags would be pre-programmed with the certification agencies public key and could therefore verify the validity of the registration in real-time. In order to prevent replay attacks from rogue readers, not only the ID of a reader, but also the public key of its owner would be signed by the agency (and subsequently transmitted to the tags), which would use this public key for all subsequent communication with the reader. Unauthorized readers would therefore also need the real owner's private key in order to decipher tag IDs. Even though certificate revocation will not work with this scheme, the damage due to unrevokable certificates seems negligible, given the ability of consumer interest groups or concerned citizens to use watchdog tags with online lookup capabilities to detect misuse. Also, certified reader IDs could allow tags to implement the resurrecting duckling model proposed by Stajano [326], where tags would only respond to a "mother" reader, but ignore requests from all others. Instead of killing tags at checkout, stores would transfer their "mother" rights to the customer's reader, thus allowing for a safe post-sales RFID usage. Additionally, such "mother" readers could inhibit replies by "its" tags for non-desired purposes and intentions by unknown readers by programming the tags accordingly.

6.5 Summary

We have argued in this section that by focusing on access control aspects alone, the problem of privacy can hardly be solved for RFID system. Instead, we propose to apply the principles developed in section 3.4 to provide *usable* privacy protection to data subjects:

- *Notice and disclosure* through embedded policy announcements within the reader-to-tag protocol.
- *Choice and consent* through an optional privacy assistant (watchdog tag), which can provide detailed information with the help of a user proxy, and support selective jamming if needed.
- *Anonymity and pseudonymity* is supported through our collection limitation mechanism, allowing data collectors to specifiy which tags are needed for a particular application, thus ignoring irrelevant tags.
- *Proximity and locality* can be explicitly expressed using collection types, indicating local and/or anonymous processing.
- *Adequate security* can be provided through the use of a privacy-aware database backend such as PawDB, while not overloading the reader-tag interface with cryptographic functions that impede usage and economic feasibility.
- *Access and recourse* is possible due to the detailed reader-policy IDs in every read attempt which provide a way to better reconstruct individual tag reads.

As we have done in the design of PawS, we assume that both social and legal norms will prompt the majority of participants in any RFID data exchange to play by the rules. The use of our proposed protocol extensions will still allow unauthorized read attempts by readers not conforming to our specification, just as PawS could not prevent hidden cameras and microphones to go undetected. While consumers carrying a watchdog tag might be able to actively jam or block the tag-to-reader communication (i.e., act like a blocker-tag [186]), for example based on user preferences regarding the reader's ID (e.g., following an online lookup), the average consumer would still need to resort to explicitly disabling her tags in order to completely prevent misuse. However, even without any additional devices, the required selection mechanism at the protocol level supports the core principle of *collection limitation*, while the compulsory identification string facilitates the principles of *openness* and *accountability*, thus providing the same level of protection as today's compulsory forms, signs, and placards announcing the privacy policy of the data collector. While they might be ignored in the routine of our everyday, their presence forms an important legal

lever once a dispute over the proper use of personal data arises. Its simplicity provides for a readily available, practical solution to many of today's RFID privacy concerns, while the possible integration of the watchdog tag functionality into future mobile phones might even make the detection of an RFID reader, its policy, and location in the future as easy as detecting the signal strength and operator IDs on a mobile phone today.

7 Summary

We are entering a time when our power to muck about with the structures that regulate is at an all time high. It is imperative, then, that we understand just what to do with this power. And, more importantly, what not to do.
Lawrence Lessig[1]

This work proposed a technical infrastructure to help allievate the challenges to privacy in a world full of ubiquitous computing services. After an extensive review of the literature addressing privacy, and an in-depth analysis of its moral and ethical aspects, the related social and legal norms, and the existing technical challenges, we provided three contributions:

- A method to announce privacy policies in smart environments via *privacy beacons* and personal *privacy assistants*;
- A method to reason and act upon such policies by automatically configuring the available services with the help of *privacy proxies*; and
- A method to store the collected information and enforce their respective collection and usage policies through *privacy-aware databases*.

In this last chapter, we want to reexamine the arguments that led us to our prototype implementation, summarize our technical approach to privacy in ubiquitous computing, outline future work in this area, and give a brief outlook of the upcoming problems and challenges ahead.

[1] In [217].

7.1 Building Privacy-Aware Ubiquitous Computing Systems

Ubiquitous computing has the potential to thoroughly change the way we live and interact, as it allows much more of our lives than ever before to be sensed, stored, accessed, and searched electronically. The proposed solutions range from giving up all (false pretexts of) privacy and create a *transparent* society in which *everybody* is accountable to anybody else [48], to comprehensive control architectures that try to embed watermarks and copy-control features into individual data elements such as email addresses and other identifiers [8, 72].

With PawS we have aimed for the middle ground: by trying to preserve today's level of privacy protection, individuals might not get "perfect" protection in the sense that it is impossible for their personal data to get involuntarily disclosed. However, for all practical purposes, such guarantees might be impossible to give, nor would these be convenient in practice. Privacy and security, afterall, are merely attributes of our daily actions and not goals in themselves: While many people whish (or expect) their actions to be as secure and private as possible, few *privacy fundamentalists* [5] are willing to actually change their behavior to achieve this.

In our chapters one through six, we thus argued instead for a system for the majority of *privacy pragmatists* and *privacy unconcerned*. We pointed out that the technology and applications of ubiquitous computing have serious implications with respect to privacy, even if one is not concerned (chapter 1). We argued that privacy is not just a question of total anonymity and security, but a complex negotiation of information flows and boundary controls (chapter 2). We showed that technical tools are only one facet of a range of mechanisms with which individuals and societies protect their privacy, in addition to moral values and norms, and legal frameworks (chapter 3). We thus presented a system that supports the Fair Information Practices in a non-intrusive, automated manner, in order to provide feedback and control mechanisms to both individuals and regulators (chapter 4). We discussed how such a concept can be integrated with a range of related technical approaches in order to provide a more comprehensive and usable system (chapter 5), and last not least presented an example of such an integration in the realm of RFID privacy (chapter 6).

7.1.1 The Case for Privacy-Aware Systems

Chapter 1 focused on motivating our work from three angels: current social and technological trends, the social implications of ubiquitous computing technology, and the lack of awareness of such issues among the designers of such systems.

We saw in section 1.1 that the vision of ubiquitous computing is primarily driven by the continuous development in microprocessors, material sciences, wireless communication technology, and sensors. However, maybe equally relevant for this continuing trend towards a future of ubiquitous computing is the fertile social environment in which this technology is applied. Efficiency, convenience, and security are areas that are in high demand and which might benefit substantially from the use of ubiquitous computing technology.

Within such a comprehensive setting as ubiquitous computing, a loss of privacy would not simply result in a few more unsolicited emails or phone calls. In section 1.2 we showed that the widespread use of ubiquitous computing systems might also result in an increased dependability (and thus vulnerability) of having the right information available at the right time and place. By having many of our routine tasks be taken care of by automated processes, we would not only run the risk of loosing control of our lives, but also giving others more control over them, as their intimate knowledge of our preferences and habits might allow them subtle ways of influencing our decisions. Last not least, ubiquitous computing systems could threaten the social integration of different parts of the population, as they not only prohibit those without the proper access technology or cognitive abilities from fully participating in society, but might also increase inequality through the widespread commercial use of detailed personal profiles – a *social sorting* [228] that witholds information and services from the economically uninteresting.

Despite this high relevance for privacy protection in ubiquitous computing systems, current research in this area is often working around privacy issues, as researchers and designers have difficulties incorporating privacy into their systems (section 1.3). The series of interviews and site visits that we conducted in 2002 and 2003 in five different projects within the EU-funded Disappearing Computer Initiative showed that those who create such system often do not feel responsible for incorporating privacy, do not think it a problem, or fear that no solution is possible.

Thus our case for tools and system support for privacy in ubiquitous computing was stated in chapter 1: Both technology and society are driving factors for systems that have far reaching consequences for our lives, especially due to the comprehensive data collection properties, yet these problems are often consciously or unconsciously overlooked by designers and researchers.

7.1.2 Conceptualizing Privacy

Chapter 2 tried to look at why it is hard to think about privacy, and help untangle the various different concepts that form the single word *privacy*. It then described how these concepts apply in the context of ubiquitous computing systems.

Section 2.1 provided three different views on the concept of privacy: facets, borders, and motivations. It described three different kinds privacy facets: procedural facets, functional facets, and constitutional facets. Procedural facets describe privacy in terms of what is to be protected, i.e., as bodily privacy, communication privacy, territorial privacy, or information privacy. Functional facets look at the effects of protecting one's privacy: zonal privacy as providing a secluded space, relational privacy for protecting our intimate and not-so-intimate relationships, and decisional privacy for protecting our ability to freely decide our lives. The constitutional approach divides privacy into the factors that provide privacy, namely solitude, anonymity, and control.

An alternative view was to look at when one feels that his or her privacy has been violated. Following the work by Marx [233], we described four *personal borders*: natural borders, social borders, spatial or temporal borders, and ephemeral borders. When personal information crosses such borders without the individual's consent, an act of privacy violation is felt.

Last not least, we looked at privacy *motivations* from a legal perspective, following Lessig [217] who lists four major driving factors for privacy legislation: empowerment, utility, dignity, and as a constraint of power. We saw how laws try to balance the each of these motivations for protecting an individual's privacy with the needs of society at large. We also described how choosing the right motivation becomes important when interpreting existing laws in light of new technology.

Section 2.2 then looked at the effect of ubiquitous computing technology on our everyday privacy, trying to explain how the combination

of invisible computers, wireless communication, efficient sensors, and smart detection and searching technology can significantly alter the level of privacy available to us. It pointed out the vastly increased collection scale of ubiquitous computing systems, which might collect data about us in many different contexts (e.g., school, office, restaurant) and times. It described the novel manner of data collections, which will happen in a much less obvious way than today's credit card transactions or web page visits, which still provide some level of feedback that some electronic trace of an action might be left behind. Ubiquitous computing systems will also collect different types of data, such as movements, walking patterns, or heart rates, which in comparison to one's email or street address require an explicit *interpretation* to be useful. The extensive reliance on context awareness will increase the motivation for collecting seemingly random bits of information, in the hope that they might be combined with other data to lead to some useful conclusions. And last not least, ubiquitous computing will also provide better routines to extract, reason on, and search such information.

Chapter 2 tried to provide us with the knowledge to evaluate existing tools and mechanisms for personal privacy (in chapter 3) by answering two questions: *What* do we mean when we talk about privacy? And *why* is privacy relevant in the context of ubiquitous computing?

7.1.3 Social, Legal, and Technical Foundations

Chapter 3 both deepened our understanding of the concept of privacy and described approaches to *solutions* for preserving our privacy. Besides reviewing existing technical privacy tools, it specifically included social and legal structures that might support (or could be supported by) a technical solution.

Section 3.1 described the various ethical disciplines – metaethics, normative ethics, comparative ethics – and illustrated how the concept of privacy could be motivated differently based on the moral approach chosen. A number of examples from applied ethics tried to demonstrate the practical impact of this, e.g., in the context of technology assessment projects. We also extensively reviewed the notion of interpersonal and institutional trust, and characterized how psychological, social, and economic aspects both influence and require trust decisions in the context of privacy. Leaving our personal information with a third party both calls for and sustains trust relationships in society, and this

section tried to show that the existing norms and ethics pemeating our daily life provide a conducive environment for such transactions.

Closely linked to our moral values are the legal frameworks described in section 3.2. We described how different cultural approaches can lead to different regulations with respect to privacy, and how even today, privacy legislation is still a highly debated issue. While most modern legislation is commonly based on the OECD's *Fair Information Practices* of 1980 [260], we illustrated how subtle shifts in interpretation lead to quite different practical implementations, resulting in the overarching privacy directive [94] in Europe and the fragmented sectorial approach and the *Safe Harbor* agreement in the US. This section also illustrated the recent tension between strong privacy legislation and law enforcement effectiveness, which could have a strong influence on any ubiquitous computing privacy solution by limiting the amount of anonymization permitted.

Section 3.3 finally gave us our technical "toolbox" in the form of encryption, authentication, anonymity, pseudonymity, transparency, and trust mechanisms. PawS builds upon a number of existing infrastructures, such as encrypted SLL connections, digital signatures, mix networks, and maybe most importantly, the P3P framework [82].

In the last part of this chapter, we set forth our own list of guiding principles, based on these social, legal, and technical mechanisms: *notice and disclosure*, *choice and consent*, *anonymity and pseudonymity*, *proximity and locality*, *adequate security*, and *access and recourse*.

Chapter 3 thus layed the foundation for our own technical infrastructure, not only in terms of technical background, but also in terms of social grounding, legal compliance, and practical principles.

7.1.4 Providing Feedback and Control

Chapter 4 described a technical infrastructure – PawS – that complements the previously described social, legal, and technical mechanisms in order to provide privacy in a ubiquitous computing service infrastructure.

Section 4.1 introduced the four parts that make up our infrastructure: *privacy contracts*, *privacy proxies*, *privacy beacons*, and our *privacy database*.

Section 4.2 described how we extend the P3P format to account for ubiquitous computing environments, such as adding perception data

and location information. It also gave an account of the mechanisms for providing remote access to collected information, and on the extended retention information present in privacy contracts.

Section 4.3 described how the privacy proxy protocol (PRO2) supports the exchange of such contracts via SOAP messages, enabling *user proxies* to request and receive contracts from *service proxies* and agree on a set of terms for a particular data collection. It also outlined how secure transport layers (SSL) and digital signatures support our notion of adequate security.

The privacy beacons described in section 4.4 form the link between the service interface and the user of a ubiquitous computing service. Signals emitted from a beacon can be intercepted by a *privacy assistant* device, supporting our principles of proximity and locality and allowing users to delegate the decision and actual act of data transfer to their personal user proxies in a seamless fashion.

The last part of the system, presented in section 4.5, is the privacy-aware database PawDB, which provides for the integral storage of both data elements and their corresponding collection policies, thus supporting the seamless enforcement of usage, retention, and recipient policies.

This chapter hence implemented our three core contributions, based on our analyses in the preceeding chapters, namely: a method to announce privacy policies in smart environments; a method to reason and act upon such policies by automatically configuring the available services; and a method to store the collected information and enforce their respective collection and usage policies.

7.1.5 Related Approaches

Chapter 5 tried to put our own contributions in context to existing work with a similar or related focus.

In section 5.1 we described related work in the areas of computational trust, privacy databases, user interfaces, and privacy infrastructures. We concluded that alternative approaches based on trust computations were inadequate for an effective and, more importantly, predictable service interaction. We described the significant challenges in building privacy databases, which we only barely touched upon in our simple demonstrator prototype of PawDB, but which offers much initiative for future work (see section 7.2 below). The same holds for the user interface problem, namely both the immediate *feedback* of current data

flows, and the means for user *control* of these data transfers. While a number of different approaches exist, research in this area is still at a very early stage. However, the few proposed privacy infrastructures in the area of ubiquitous computing all more or less follow the guidelines and general outline of our PawS publications [205, 206], though with a slightly different focus: Hong et al.'s Confab Toolkit [167] addresses primarily interpersonal data exchange, while Myles et al. explicitly incorporate a location server into the system.

We then presented a summary of the field of location privacy in section 5.2. We described that ubiquitous computing systems collected location information both intentionally (as part of a location-based service) and unintentionally (as part of a localized service), and that location privacy not only attempts to prevent eavesdropping or otherwise illegally acquiring personal location data, but also to minimize the information leakage of location or identification data. We pointed out that PawS is well suited to minimize information leakage, though specialized approaches such as automated location obfuscation [145] might further improve this process for specific applications. Of particular importance is the fact that even pseudonymously collected location information has a high potential for deanonymization, as combining such information with other, personalized data sources (e.g., office addresses or hobbies) can often yield high identification rates. While limiting location information disclosure can also be achieved within our PawS framework, the section reinforced our analysis in section 2.2.3 with respect to the difficulty of anonymizing perception data in ubiquitous computing systems.

The section closed with a detailed discussion of the current approaches to RFID privacy in section 5.3. After extensively reviewing existing approaches to RFID privacy – including for example the kill-command, hash locks, metaIDs, and the Blocker-Tag – we concluded that most focus too narrowly on security aspects of the system, e.g., trying to prevent unauthorized reads. This results in systems that shift the burden of protection onto the consumer, who, in the manner of Hobbes "war of all against all" [130], is expected to indiviudally fight back against otherwise unbound third parties.

7.1.6 PawS and RFID privacy

Drawing support from our social and legal analysis in chapter 3, we instead proposed a *transparency protocol* for RFID systems that incorporates our announcement mechanism directly into the reader-to-tag protocol. Chapter 6 described this approach in detail, which allows both consumers and interested parties to easily detect illegal or non-conforming read attempts, while leaving both the definition and the enforcement of acceptable read practices to society and legal frameworks.

After briefly describing a stock RFID protocol in section 6.1 on which we based our extensions on, we outlined the main parts of our system in section 6.2: the *Reader-Policy-ID* (RPID) to identify collectors, policies, and their readers; the *Purpose Specification* to better describe the purpose of a tag read; the *Collection Types* in order to differentiate between local, anonymous, and identifying reads; and our *Tag Selection* mechanism that allows application-specific collection limitations.

Section 6.3 presented our *Watchdog Tag* prototype, a PDA that allows individuals to detect and display the information present in the reader-to-tag protocol, similar to our *privacy assistant* in PawS.

We closed the chapter with a feasibility discussion that showed the practicality of such an approach, both in terms of read efficiency and deployment costs (section 6.4), as well as pointing out areas of future work, such as an improved security infrastructure based on reader certificates.

7.2 Future Work

Our work provided a first attempt at providing an automated privacy solution for services in ubiquitous computing environments. As discussed in section 4.6 above, our experiences with conceiving and implementing such a system suggest several avenues for further work.

7.2.1 Beacon Announcements

Our prototype beacons use infrared to emit policy announcements. While this has the advantage of providing *locality* to the system (i.e., announcements are locally scoped, typically on a per room basis), this also requires a line of sight to the user's privacy assistant. Other wireless communication protocols such as WLAN, Bluetooth, or Zigbee

could operate under much more restricted conditions, e.g., if the user's privacy assistant is part of his or her (potentially covered) clothing, such as a wristwatch, or stowed away in a bag, in case of a mobile phone. For such scenarios, the system would benefit from a localization feature that would allow service proxies to express the reach of their advertised sensors, and privacy assistants to determine whether their current position would be covered by such sensors (e.g., a camera system). At the same time, however, such a solution would also increase the complexity of the system, as policy announcements would need to explicitly include a location model of their applicability.

7.2.2 Database Implementation

While our current PawDB implementation supports the storage and usage of collected information according to the agreed-upon privacy contracts, it lacks both the efficiency needed for real-world usage and the interoperability with subsequent privacy-aware databases, e.g., of a third party providing some service fulfillment on behalf of the original data collector. As the work of Agrawal et al. [11] has pointed out, a more efficient implementation of our concepts is possible. However, for a comprehensive solution, such a database would still need to provide better anonymization support such as the work presented by Sweeney [330] in order to better prevent service providers from inadvertly deanonymizing pseudonymized data. Also, in the manner of Karjoth et al.'s "E-P3P" work [192], data transmissions between several independant privacy-aware databases would need standard semantics in order to properly transform the original privacy contract into a derived third-party contract. Last not least, future PawDB systems could also more explicitly support the principle of *locality and proximity*, e.g., by forcing queries to be issued from a physically close location from where the data was originally located.

7.2.3 User Interface

The user interface is probably the most crucial feature of any such system. While we previously pointed to alternative models such as the *faces* metaphor [214] or identity management systems [180], recent research has demonstrated that even those approaches might lead to unexpected pitfalls. In follow-up work to their *faces* system, Lederer et al. [215] identified two main problems of current privacy interface ap-

proaches (including the previous work of their own): failure to provide users with a proper *understanding* of his or her data flows, and lack of support to conduct socially meaningful *actions* through the provided interface.[2]

7.2.4 Mechanisms for Peer Privacy

We already pointed out in our previous chapter that PawS does not attempt to regulate information flows between peers, e.g., between friends that allow each other to track their location over the course of the day, or between family members that transmit their home activity to each other via something like the family portrait [250]. However, it might be beneficial to unify these two diverse approaches – organizational privacy through broadcasted privacy policies, and interpersonal privacy through mechanisms of obfuscation and plausible deniability – into a single privacy framework. Obviously, different applications might require very different affordances, e.g., using a public print server compared to the sharing of one's household activity with a (physically) distant relative.

7.3 Outlook

An often cited bon mot states "*Predictions are tough, especially when they concern the future.*" This applies equally well to the topic of this thesis – the development of privacy protection in future ubiquitous computing environments. What does seem clear is the relentless technological progress that will eventually make smart coffee cups, smart shirts, smart homes, or even smart dust technically feasible – though maybe neither economically viable nor individually desirable. We also might assume that the digitalization of our everyday, i.e., the codification of our daily actions into machine-readable symbols and their subsequent storage and processing, will continue to apply to an ever large share of our lives. This is because of the three societal trends that we mentioned in chapter 1:

- The constant thrive of society to have machines lighten the burden of everyday life – from cleaning dishes to automatically ordering groceries with a smart fridge.

[2] In Bellotti and Sellen's terms [33] this would be *feedback* and *control*.

- The increased economic benefits of monitoring goods in real time [120] or automatically determining the best price for a product in real-time [254].
- The heightened security concerns after September 11 that prompt society to demand not only an increased apprehension rate but also better pre-emptive detection and crime prevention.

In each of these areas, privacy gets in the way of a more convenient, efficient, and potentially safer life. It is ultimately up to society to decide what level of privacy it deems appropriate in a future where so much data about each individual can be collected. Lawmakers, on the other hand, need to revise existing laws to take these new kinds of ubiquitous data collections into account, to both relax today's often too rigid statues, and to provide clear guidelines regarding the potential for creating identifiable data from a number of different anonymous data sets [300]. Technologists, then, should take care to create systems that do indeed leave such decisions to society by making privacy a viable option to choose.

Bibliography

[1] Gregory D. Abowd. Classroom 2000: An experiment with the instumentation of a living educational envrionment. *IBM Systems Journal*, 38(4):508–530, October 1999. Available from: `www.research.ibm.com/journal/sj/384/abowd.html`.

[2] Gregory D. Abowd, Barry Brumitt, and Steven A. Shafer, editors. *Proceedings of the Third International Conference on Ubiquitous Computing (UbiComp 2001)*, number 2201 in LNCS, Atlanta, USA, September 2001. Springer-Verlag.

[3] Gregory D. Abowd and Elizabeth D. Mynatt. Charting past, present and future research in ubiquitous computing. *ACM Transactions on Computer-Human Interaction, Special issue on HCI in the new Millenium*, 7(1):29–58, March 2000.

[4] Mark S. Ackerman. Privacy in pervasive environments: Next generation labeling protocols. *Personal and Ubiquitous Computing*, 8(6):430–439, September 2004.

[5] Mark S. Ackerman, Lorrie Faith Cranor, and Joseph Reagle. Privacy in e-commerce: examining user scenarios and privacy preferences. In *Proceedings of the First ACM Conference on Electronic Commerce (EC-99)*, pages 1–8. ACM Press, November 1999.

[6] Natascha Adamowsky. Kulturelle Relevanz. Ladenburger Diskurs "Ubiquitous Computing", February 2000. Available from: `www.inf.ethz.ch/vs/events/slides/adamldbg.pdf`.

[7] Carlisle Adams and Steve Lloyd. *Understanding PKI: Concepts, Standards, and Deployment Considerations*. Addison-Wesley Publishing, Boston, 2002. Chapter *Core PKI Services* available at `www.microsoft.com/technet/security/topics/identity/corepki.mspx`.

[8] Gagan Aggarwal, Mayank Bawa, Prasanna Ganesan, Hector Garcia-Molina, Krishnaram Kenthapadi, Nina Mishra, Rajeev Motwani, Utkarsh Srivastava, Dilys Thomas, Jennifer Widom, and Ying Xu. Vision paper: Enabling privacy for the paranoids. In Mario A. Nascimento, M. Tamer Özsu, Donald Kossmann, Renée J. Miller, José A. Blakeley, and Berni Schiefer, editors, *Proceedings of the Thirtieth International Conference on Very Large Data Bases (VLDB 2004)*, pages 708–719, Toronto, Canada, 2004. Morgan Kaufmann Publishers. Available from: `www.vldb04.org/protected/eProceedings/`.

[9] Rakesh Agrawal, Jerry Kiernan, Ramakrishnan Srikant, and Yirong Xu. Hippocratic databases. In *Proceedings of the 28th International Conference on Very Large Databases (VLDB 2002)*, pages 143–154, Hong Kong, August 2002. Morgan Kaufmann. Available from: `www.vldb.org/conf/2002/S05P02.pdf`.

[10] Rakesh Agrawal, Jerry Kiernan, Ramakrishnan Srikant, and Yirong Xu. Server-centric P3P. W3C Workshop on the Future of P3P, November 2002. Available from: `www.w3.org/2002/p3p-ws/pp/Overview.html`.

[11] Rakesh Agrawal, Jerry Kiernan, Ramakrishnan Srikant, and Yirong Xu. Implementing P3P using database technology. In *Proceedings of the IEEE 19th International Conference on Data Engineering*, pages 595–606, Bangalor, India, March 2003. Computer Society, IEEE Press. Available from: `ieeexplore.ieee.org/xpl/RecentCon.jsp?puNumber=8910`.

[12] Rakesh Agrawal, Jerry Kiernan, Ramakrishnan Srikant, and Yirong Xu. An Xpath-based preference language for P3P. In *Proceedings of the 12th International World Wide Web Conference (WWW2003)*, pages 629–639, Budapest, Hungary, May 2003. ACM Press. Available from: `portal.acm.org/citation.cfm?id=775241&jmp=cit&dl=portal&dl=ACM`.

[13] Rakesh Agrawal and Ramakrishnan Srikant. Privacy-preserving data mining. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 439–450, Dallas, USA, May 2000. CM Press. Available from: `www.almaden.ibm.com/cs/people/srikant/papers/sigmod00.pdf`.

[14] Philip E. Agre and Marc Rotenberg, editors. *Technology and Privacy: The New Landscape*. The MIT Press, Cambridge, MA, USA, 1998.

[15] Jari Ahola. Ambient Intelligence. *ERCIM News*, (47), October 2001.

[16] Katherine Albrecht. Supermarket cards: The tip of the retail surveillance iceberg. *Denver University Law Review*, 79(4):534–539, 558–565, October 2002. Available from: `www.nocards.org/AutoID/overview.shtml`.

[17] Auto-ID Center/EPCglobal, Cambridge, MA, USA. *860 MHz-930 MHz Class 1 Radio Frequency (RF) Identification Tag Radio Frequency & Logical Communication Interface Specification*, 2002. Available from: `www.epcglobalinc.org/standards_technology/Secure/v1.0/UHF-class1.pdf`.

[18] Auto-ID Center/EPCglobal, Cambridge, MA, USA. *900 MHz Class 0 Radio Frequency (RF) Identification Tag Specification*, 2003. Available from: `www.epcglobalinc.org/standards_technology/Secure/v1.0/UHF-class0.pdf`.

[19] Gildas Avoine. Privacy issues in RFID banknotes protection schemes. In Jean-Jacques Quisquater, Pierre Paradinas, Yves Deswarte, and Anas Abou El Kalam, editors, *Proceedings of the Sixth International Conference on Smart Card Research and Advanced Applications (CARDIS 2004)*, pages 33–48, Toulouse, France, August 2004. Kluwer. Available from: `lasecwww.epfl.ch/~gavoine/download/avoine-cardis-banknote-paper.pdf`.

[20] Robert Axelrod. *The Evolution of Cooperation*. Basic Books, New York, USA, 1984. As cited in [70].

[21] Erik Baard. Buying trouble – your grocery list could spark a terror probe. *The Village Voice*, July 24–30, 2002. Available from: `www.villagevoice.com/issues/0230/baard.php`.

[22] Annette C. Baier. Trust and its vulnerabilities. In Annette C. Baier, editor, *Moral Prejudices*. Harvard University Press, Cambridge, MA, USA, 1995. As cited in [202].

[23] Annette C. Baier. Vertrauen und seine Grenzen. In Hartmann and Offe [158], chapter 1, pages 37–84.

[24] K. Suzanne Barber, Karen Fullam, and Joonoo Kim. Challenges for trust, fraud and deception research in multi-agent systems. In Rino Falcone, Suzanne Barber, Larry Korba, and Munindar Singh, editors, *Proceedings of the The First International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2002)*, volume 2631 of *Lecture Notes in Artificial Intelligence*, pages 8–14, Bologna, Italy, July 2002. Springer-Verlag.

[25] John Perry Barlow. A declaration of the independence of cyberspace. EFF Website. Available from: `homes.eff.org/~barlow/Declaration-Final.html`.

[26] Adam Barnhart. Erving goffman: The presentation of self in everyday life. Online Essay, 1994. Available from: `www.cfmc.com/adamb/writings/goffman.htm`.

[27] Mark Bartel, John Boyer, Donald Eastlake, Barb Fox, Brian LaMacchia, Joseph Reagle, Ed Simon, and David Solo. XML-signature syntax and processing. W3C Recommendation, World Wide Web Consortium (W3C), February 2002. Available from: `www.w3.org/TR/xmldsig-core/`.

[28] Ben F. Barton and Mathalee S. Barton. Modes of power in technical and professional visuals. *Journal of Business and Technical Communication*, 7(1):138–162, 1993.

[29] Jörg Baus, Antonio Krüger, and Wolfgang Wahlster. A resource-adaptive mobile navigation system. In *IUI '02: Proceedings of the 7th international conference on Intelligent user interfaces*, pages 15–22. ACM Press, 2002.

[30] Google's Gmail sparks privacy row. *BBC News, World Edition*, April 5, 2004. Available from: `news.bbc.co.uk/2/hi/business/3602745.stm`.

[31] Tom L. Beauchamp. Methods and principles in biomedical ethics. *Journal of Medical Ethics*, 29(5):269–274, 2003.

[32] Victoria Bellotti. Privacy and multimedia. In Agre and Rotenberg [14], chapter 2, pages 63–98.

[33] Victoria Bellotti and Abigail Sellen. Design for privacy in ubiquitous computing environments. In *Proc. of the Third European Conference on Computer-Supported Cooperative Work (ESCW '93)*, pages 77–92. Kluwer, 1993.

[34] Benetton. No microchips present in garments on sale – no decision yet taken on industrial use. Press Release, April 4, 2003. Available from: `www.benetton.com/press/sito/_media/press_releases/rfiding.pdf`.

[35] Alastair R. Beresford and Frank Stajano. Location privacy in pervasive computing. *IEEE Pervasive Computing*, 2(1):46–55, January 2003. Available from: `www-lce.eng.cam.ac.uk/~arb33/papers/BeresfordStajano-LocationPrivacy-IEEEPervasive2003.pdf`.

[36] Alastair R. Beresford and Frank Stajano. Mix zones: User privacy in location-aware services. First IEEE International Workshop on Pervasive Computing and Communication Security (PerSec'04), March 2004. Available from: `www-lce.eng.cam.ac.uk/~arb33/papers/BeresfordStajano-MixZones-PerSec2004.pdf`.

[37] Isaiah Berlin. *Four Essays on Liberty*. Oxford University Press, Oxford, UK, 1969.

[38] Matt Blaze, Joan Feigenbaum, and Jack Lacy. Decentralized trust management. In *Proceedings of the 1996 IEEE Symposium on Security and Privacy*, pages 164–173, Oakland, CA, USA, May 1996. Available from: `citeseer.nj.nec.com/blaze96decentralized.html`.

[39] Jürgen Bohn, Vlad Coroamă, Marc Langheinrich, Friedemann Mattern, and Michael Rohs. Living in a world of smart everyday objects – social, economic, and ethical implications. *Journal of Human and Ecological Risk Assessment*, 10(5):763–786, October 2004.

[40] Jürgen Bohn, Vlad Coroamă, Marc Langheinrich, Friedemann Mattern, and Michael Rohs. Social, economic, and ethical implications of ambient intelligence and ubiquitous computing. In Werner Weber, Jan Rabaey, and Emile Aarts, editors, *Ambient Intelligence*. Springer-Verlag, 2004.

[41] Friedel Bolle. Does trust pay? Discussion paper, Europa-University Viadrina, Frankfurt (Oder), Germany, 1998. Available from: viadrina.euv-frankfurt-o.de/~vwlmikro/veroeffentlichungen/bolle/does_trust_pay.pdf.

[42] Detlef Borchers. Frischkäse bitte bei Kasse 3 melden – Funketiketten wecken diffuse Ängste. *Neue Zürcher Zeitung*, (54):63, March 5, 2004. Available from: www.nzz.ch/2004/03/05/em/page-article9G4V4.html.

[43] John J. Borking. Der Identity Protector. *DuD – Datenschutz und Datensicherheit*, 96(11):654–658, 1996. Available from: www.datenschutzzentrum.de/somak/somak96/sa96bork.htm.

[44] Jan Bormans and Keith Hill. Mpeg-21 overview. ISO/IEC JTC1/SC29/WG11 N5231, International Organisation for Standardization, Shanghai, China, 2002. Available from: www.chiariglione.org/mpeg/standards/mpeg-21/mpeg-21.htm.

[45] Stowe Boyd. The state of social tools. *Darwin Magazine*, June 2004. Available from: www.darwinmag.com/read/060104/boyd.html.

[46] Andrew Brandt. A little bird that guards your online privacy. *PC World*, 20(12), December 2002. Available from: www.pcworld.com/howto/article/0,aid,105830,00.asp.

[47] Jack Brassil. Using mobile communications to assert privacy from video surveillance. In *Proceedings of the First International Workshop on Systems and Network Security (SNS2005)*, Denver, USA, 2005. IEEE, IEEE Press. Available from: www.hpl.hp.com/personal/Jack_Brassil/cloak.pdf.

[48] David Brin. *The Transparent Society*. Perseus Books, Reading MA, 1998.

[49] Henry Peter Brougham. *Historical Sketches of Statesmen Who Flourished in the Time of George III*, volume 1. Lea & Blanchard, Philadephia, PA, USA, 1839. As quoted in [265].

[50] Allen Brown, Barbara Fox, Satoshi Hada, Brian LaMacchia, and Hiroshi Maruyama. SOAP security extensions: Digital signature. W3c note, World Wide Web Consortium (W3C), February 2001. Available from: www.w3.org/TR/SOAP-dsig.

[51] John G. Bruhn. *Trust and the Health of Organizations*. Kluwer Academic, New York, USA, 2001.

[52] Herbert Burkert. Privacy-enhancing technologies. In Agre and Rotenberg [14], chapter 4, pages 125–142.

[53] Andreas Butz, Clifford Beshers, and Steven Feiner. Of vampire mirrors and privacy lamps: privacy management in multi-user augmented environments. In *UIST '98: Proceedings of the 11th annual ACM symposium on User interface software and technology*, pages 171–172. ACM Press, 1998.

[54] Jan Camenisch and Els Van Herreweghen. Design and implementation of the idemix anonymous credential system. In *Proceedings of the 2002 ACM Conference on Computer and Communications Security*, pages 21–30, 2002.

[55] Jean L. Camp and Stephen Lewis, editors. *Economics of Information Security*, volume 12 of *Advances in Information Security*. Springer-Verlag, 2004.

[56] Kathy Cartrysse, Ricardo Corin, Marnix Dekker, Sandro Etalle, Jaap-Henk Hoepman, Gabriele Lenzini, Jan v.d. Lubbe, Jan Verschuren, and This Veugen. Privacy in an ambient world (PAW) – using licenses and private computing as PET. Position paper, University of Nijmegen, Nijmegen, The Netherlands, 2004. Available from: `www.cs.ru.nl/paw/`.

[57] Fred H. Cate. *Privacy in the Information Age*. The Brookings Institution, Washington, D.C., USA, online edition, 1997. Available from: `brookings.nap.edu/books/0815713169/html`.

[58] Jason Catlett. Open letter to P3P developers and replies. In *Proceedings of the Tenth Conference on Computers, Freedom, and Privacy: Challenging the Assumption (CFP 2000)*, pages 157–164, Toronto, Canada, 2000. Available from: `www.junkbusters.com/standards.html`.

[59] CDT – Center for Democracy & Technology. Summary of H.R. 5018, the Electronic Communications Privacy Act of 2000, September 2000. Available from: `www.cdt.org/security/000927hr5018.shtml`.

[60] David Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2):84–88, February 1981. Available from: `world.std.com/~franl/crypto/chaum-acm-1981.html`.

[61] David Chaum, Amos Fiat, and Moni Naor. Untraceable electronic cash. In *CRYPTO '88: Proceedings on Advances in Cryptology*, pages 319–327. Springer-Verlag, 1990.

[62] Lifestyle: Chipping away at your privacy. *Chicago Sun-Times*, November 9, 2003. Available from: `www.suntimes.com/output/lifestyles/cst-nws-spy09.html`.

[63] Roger Clarke. P3P re-visited. *Privacy Law & Policy Reporter*, 8(4):81–83, 2001. Available from: `www.austlii.edu.au/au/journals/PLPR/2001/39.html`.

[64] Networking: Gillette shrugs off RFID-tracking fears. *C|Net News.com*, August 14, 2003. Available from: `news.com.com/2100-1039_3-5063990.html?tag=cd_mh`.

[65] Radio ID chips may track banknotes. *C|Net News.com*, May 22, 2003. Available from: `news.com.com/news.com.com/2100-1017_3-1009155.html`.

[66] Vary Coates. Commentary: Can the office for technology assessment be privatized? *The Scientist*, 10(2), January 1996. Available from: `www.the-scientist.com/yr1996/jan/ota_960122.html`.

[67] Peter Cochrane. Head to head. *Sovereign Magazine*, pages 56–57, 2000. Available from: `www.cochrane.org.uk/opinion/papers/prof.htm`.

[68] Julie E. Cohen. Examined lives: Informational privacy and the subject as object. *Stanford Law Review*, 52:1373–1437, May 2000. As cited in [322]. Available from: `www.law.georgetown.edu/faculty/jec/examined.pdf`.

[69] James S. Coleman. *Foundations of Social Theory*. Belknap Press, Cambridge, USA, 1990.

[70] James S. Coleman and Thomas J. Fararo. *Rational Choice Theory: advocacy and critique*. Sage Publications, Newbury Park, USA, 1992.

[71] Privacy concerns as Benetton adds "smart tags" to clothing line. *ComputerWeekly.com*, March 13, 2003. Available from: `www.computerweekly.com/Article120113.htm`.

[72] Claudine Conrado, Milan Perković, and Willem Jonker. Privacy-preserving digital rights management. In Willem Jonker and Milan Petković, editors, *Proceedings of the First International Workshop on Secure Data Management (SDM 2004) at VLDB 2004*, number 3178 in LNCS, pages 83–99, Toronto, Canada, 2004. Springer-Verlag.

[73] ContenGuard Inc. Xrml 2.0. Technical overview, March 2002. Available from: `www.xrml.org/Reference/XrMLTechnicalOverviewV1.pdf`.

[74] Vlad Coroamă and Felix Röthenbacher. The chatty environment – providing everyday independence to the visually impaired. The 2nd International Workshop on Ubiquitous Computing for Pervasive Healthcare Applications (UbiHealth2003), October 2003. Available from: `www.inf.ethz.ch/vs/publ/papers/ubicomp2003-hc.pdf`.

[75] Council of Europe. Convention for the protection of human rights and fundamental freedoms. CETS 005, November 1950. Available from: `conventions.coe.int/Treaty/en/Treaties/Html/005.htm`.

[76] Council of Europe. Resolution (73) 22 on the protection of the privacy of individuals vis-à-vis electronic data banks in the private sector, 1973. Available from: `www.coe.int/T/E/Legal_affairs/Legal_co-operation/Data_protection/Documents/International_legal_instruments/Resolution%20(73)%2022.asp`.

[77] Council of Europe. Resolution (74) 29 on the protection of the privacy of individuals vis-à-vis electronic data banks in the public sector, 1974. Available from: `www.coe.int/T/E/Legal_affairs/Legal_co-operation/`

Data_protection/Documents/International_legal_instruments/Resolution%20(74)%2029.asp.

[78] Council of Europe. Convention for the protection of individuals with regard to automatic processing of personal data. CETS 108, January 1981. Available from: conventions.coe.int/Treaty/en/Treaties/Html/108.htm.

[79] Lorrie Cranor, Brooks Dobbs, Serge Egelmann, Giles Hogben, Jack Humphrey, Matthias Schunter, David A. Stampley, and Rigo Wenning. The platform for privacy preferences 1.1 (P3P1.1) specification. W3C Working Draft, World Wide Web Consortium, January 2005. Available from: www.w3.org/TR/P3P11/.

[80] Lorrie Cranor, Marc Langheinrich, and Massimo Marchiori. A P3P preference exchange language 1.0 (APPEL1.0). W3C Working Draft, April 2001. Available from: www.w3.org/TR/P3P-preferences.

[81] Lorrie Cranor, Marc Langheinrich, Massimo Marchiori, and Joseph Reagle. The platform for privacy preferences 1.0 (P3P1.0) specification. W3C Recommendation, World Wide Web Consortium, April 2002. Available from: www.w3.org/TR/P3P/.

[82] Lorrie Faith Cranor. *Web Privacy with P3P*. O'Reilly & Associates, Sebastopol, USA, 2002. Available from: www.p3pbook.com.

[83] Lorrie Faith Cranor and Jr. Joseph Reagle. Designing a social protocol: Lessons learned from the platform for privacy preferences project. In *Proceedings of the Telecommunications Policy Research Conference*, Alexandria, USA, September 1997. Available from: lorrie.cranor.org/pubs/dsp/.

[84] Lorrie Faith Cranor, Joseph Reagle, and Mark S. Ackerman. Beyond concern: Understanding net users' attitudes about online privacy. Technical Report TR 99.4.3, AT&T Labs-Research, April 1999. Available from: www.research.att.com/library/trs/TRs/99/99.4/.

[85] Jorge R. Cuellar, John B. Morris, Deirdre Mulligan, Jon Peterson, and James M. Polk. Geopriv requirements. RFC 3693, IETF, 2004. Available from: www.ietf.org/rfc/rfc3693.txt.

[86] Michelle Engelhardt Danley, Deirdre Mulligan, John B. Morris, and Jon Peterson. Threat analysis of the Geopriv protocol. RFC 3694, IETF, 2004. Available from: www.ietf.org/rfc/rfc3694.txt.

[87] Morton D. Davis. Game theory. In *Encyclopædia Britannica*. Deluxe Edition CD-ROM, 2004.

[88] James DeFillipis. The myth of social captial in community development. *Housing Policy Debate*, 12(4):781–806, 2001. Available from: www.fanniemaefoundation.org/programs/hpd/pdf/HPD_1204_defilippis.pdf.

[89] Morton Deutsch. Trust and suspicion. *Conflict Resolution*, 2(4):265–279, 1958.

[90] Tim Dierks and Christopher Allen. The tls protocol – version 1.0. RFC 2246, IETF, January 1999. Available from: www.ietf.org/rfc/rfc2246.txt.

[91] Whitfield Diffie and Martin E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, November 1976.

[92] Steve Ditlea. The electronic paper chase. *Scientific American*, 285(5):50–55, November 2001. Available from: www.sciam.com/article.cfm?articleID=0004C2D2-B938-1CD6-B4A8809EC588EEDF.

[93] Jos Dumortier and Caroline Goemans. Frameworks for privacy in the on-line environment. *The IPTS Report*, (42), March 2000. Available from: www.jrc.es/home/report/english/articles/vol42/welcome.htm.

[94] EC – European Commission. Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data. *Official Journal of the European Communities*, L 281(395L0046):31–50, November 1995. Available from: europa.eu.int/smartapi/cgi/sga_doc?smartapi!celexapi!prod!CELEXnumdoc&lg=EN&numdoc=31995L0046&model=guichett.

[95] EC – European Commission. Directive 1999/93/EC of the European Parliament and of the Council of 13 December 1999 on a community framework for the electronic signatures (directive on electronic signatures). *Official Journal of the European Communities*, L 13:12–20, January 2000. Available from: www.europa.eu.int/smartapi/cgi/sga_doc?smartapi!celexplus!prod!CELEXnumdoc&lg=en&numdoc=31999L0093.

[96] EC – European Commission. Directive 2002/58/EC of the European Parliament and of the Council of 12 July 2002 concerning the processing of personal data and the protection of privacy in the electronic communications sector (directive on privacy and electronic communications). *Official Journal of the European Communities*, L 201:37–47, July 2002. Available from: europa.eu.int/smartapi/cgi/sga_doc?smartapi!celexapi!prod!CELEXnumdoc&lg=en&numdoc=32002L0058&model=guichett.

[97] Wayne W. Eckerson. Data warehousing special report: Data quality and the bottom line. *ADTmag – Application Development Trends*, May 2002. Available from: www.adtmag.com/print.asp?id=6321.

[98] Science and technology: Where's the smart money? *The Economist*, February 9–15, 2002. Available from: www.economist.com/printedition/index.cfm?d=20020209.

[99] Semiconductors: Benetton backs off RFID deployment. *EETimes*, April 5, 2003. Available from: www.eetimes.com/semi/news/OEG20030405S0001.

[100] EFF – Electronic Frontier Foundation. The USA PATRIOT Act. EFF Hot Topics. Available from: www.eff.org/Privacy/Surveillance/Terrorism/PATRIOT/.

[101] EFF – Electronic Frontier Foundation. CAPPS II: Government surveillance via passenger profiling – backgrounder, 2003. Available from: www.eff.org/Privacy/cappsii/background.php.

[102] Martin Endreß. Vertrauen und Vertrautheit – Phänomenologisch-anthropologische Grundlegung. In Hartmann and Offe [158], pages 161–203.

[103] Colin English, Paddy Nixon, Sotirios Terzis, Andrew McGettrick, and Helen Lowe. Dynamic trust models for ubiquitous computing environments. Workshop on Security in Ubiquitous Computing at UbiComp 2002, October 2002. Available from: `www.teco.edu/~philip/ubicomp2002ws/`.

[104] EP – European Parliament. Charter of fundamental rights of the European Union. *Official Journal of the European Communities*, C 364(01), November 2000. Available from: `www.europarl.eu.int/charter/pdf/text_en.pdf`.

[105] EPCglobal. EPC Tag Data Specification 1.1. EPCglobal Standard, November 2003. Available from: `www.epcglobalinc.org/standards_technology/EPCTagDataSpecification11rev124.pdf`.

[106] EPIC – Electronic Privacy Information Center. The Children's Online Privacy Protection Act. The A to Z's of Privacy Website. Available from: `www.epic.org/privacy/kids/`.

[107] EPIC – Electronic Privacy Information Center. The Drivers Privacy Protection Act. The A to Z's of Privacy Website. Available from: `www.epic.org/privacy/dppa/`.

[108] EPIC – Electronic Privacy Information Center. The Gramm-Leach-Bliley Act. The A to Z's of Privacy Website. Available from: `www.epic.org/privacy/glba/`.

[109] EPIC – Electronic Privacy Information Center. The Terrorism Information Awareness (TIA) Page. The A to Z's of Privacy Website. Available from: `www.epic.org/privacy/profiling/tia/`.

[110] EPIC – Electronic Privacy Information Center. The Video Privacy Protection Act. The A to Z's of Privacy Website. Available from: `www.epic.org/privacy/vppa/`.

[111] Amitai Etzioni. *The Limits of Privacy*. Basic Books, New York, USA, 1999.

[112] Amitai Etzioni. *Next: The Road to the Good Society*. Basic Books, New York, USA, 2001.

[113] Davic C. Fallside and Priscilla Walmsley. XML schema part 0: Primer, second edition. W3c recommendation, World Wide Web Consortium (W3C), 2004. Available from: `www.w3.org/TR/xmlschema-0/`.

[114] Federal Trade Commission. Children's Online Privacy Protection Act of 1998 (COPPA), 1998. Available from: `www.ftc.gov/ogc/coppa1.htm`.

[115] Federal Trade Commission. Electronic Signatures in Global and National Commerce Act (E-Sign), 2001. Available from: `www.ftc.gov/os/2001/06/esign7.htm`.

[116] Roy Fielding, Jim Gettys, Jeffrey C. Mogul, Henrik Frystyk Nielsen, Larry Masinter, Paul J. Leach, and Tim Berners-Lee. Hypertext transfer protocol – http1.1. RFC 2616, IETF, June 1999. Available from: `www.ietf.org/rfc/rfc2616.txt`.

[117] Klaus Finkenzeller. *RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification.* John Wiley & Sons, Ltd, 2003.

[118] Kenneth P. Fishkin and Sumit Roy. Enhancing RFID privacy through antenna energy analysis. MIT RFID Privacy Workshop, Cambridge, MA, USA, November 15, 2004. Available from: `www.rfidprivacy.org/papers/fishkin.pdf`.

[119] David H. Flaherty. *Protecting Privacy in Surveillance Societies: The Federal Republic of Germany, Sweden, France, Canada, and the United States.* University of North Carolina Press, Chapel Hill, NC, USA, 1989.

[120] Elgar Fleisch, Friedemann Mattern, and Stephan Billinger. Betriebswirtschaftliche Applikationen des Ubiquitous Computing - Beispiele, Bausteine und Nutzenpotentiale. In Heinz Sauerburger, editor, *Ubiquitous Computing*, number 229 in HMD – Praxis der Wirtschaftsinformatik, pages 5–15. dpunkt.verlag, February 2003.

[121] Daniela Florescu and Donald Kossmann. Storing and querying XML data using an RDBMS. *IEEE Data Engineering Bulletin*, 22(3):27–34, September 1999.

[122] Christian Flörkemeier. Die Technologiestandards des Auto-ID Centers. In Elgar Fleisch and Friedemann Mattern, editors, *Das Internet der Dinge – Ubiquitous Computing und RFID in der Praxis*. Springer-Verlag, 2005.

[123] Christian Flörkemeier, Roland Schneider, and Marc Langheinrich. Scanning with a purpose – supporting the fair information principles in rfid protocols. In *2nd International Symposium on Ubiquitous Computing Systems (UCS 2004)*, Tokyo, Japan, November 2004.

[124] FoeBuD deckt auf: Versteckte RFID in Metro-Payback-Kundenkarte, 2003. Available from: `www.foebud.org/texte/aktion/rfid/`.

[125] Michel Foucault. *Discipline and Punish: The Birth of the Prison*. Vintage Books, New York, USA, 1995.

[126] Alan O. Freier, Philip Karlton, and Paul C. Kocher. The ssl protocol – version 3.0. Internet-draft, IETF, November 1996.

[127] Francis Fukuyama. *Trust: The Social Virtues and the Creation of Prosperity*. The Free Press, New York, USA, 1995.

[128] E. Gabber, P. Gibbons, Y. Matias, and A. Mayer. How to make personalized web browsing simple, secure, and anonymous. In *Proceedings of Financial Cryptography 97*, volume 1318 of *LNCS*. Springer-Verlag, 1997. Available from: `www.bell-labs.com/project/lpwa/papers.html`.

[129] Diego Gambetta, editor. *Trust: Making and Breaking Cooperative Relations (Electronic Edition)*. Department of Sociology, University of Oxford, UK, 2000. Available from: `www.sociology.ox.ac.uk/papers/trustbook.html`.

[130] John C. A. Gaskin, editor. *Thomas Hobbes: Leviathan*. Oxford Paperbacks, 1998.

[131] Ruth Gavison. Privacy and the limits of the law. *Yale Law Journal*, 89:421–471, 1980.

[132] Ruth Gavison. Privacy and the limits of the law. In Ferdinand D. Schoeman, editor, *Philosophical Dimensions of Privacy: An An-*

thology, pages 346–402. Cambridge University Press, Cambridge, UK, 1984. Originally published in [131] and reprinted as [133].

[133] Ruth Gavison. Privacy and the limits of the law. In Deborah G. Johnson and Helen Nissenbaum, editors, *Computers, ethics & social values*, pages 332–351. Prentice-Hall, Inc., 1995.

[134] Hans Werner Gellersen, editor. *Handheld and Ubiquitous Computing*, volume 1707 of *Lecture Notes in Computer Science*, Karlsruhe, Germany, October 1999. Springer-Verlag.

[135] Data Warehouse, Data Mining und Datenschutz. 59th Conference of Privacy-Commissioners in Germany, March 2000. Available from: `www.datenschutz-berlin.de/doc/de/konf/59/datawa.htm`.

[136] Bernhard Gert. *Morality*. Oxford University Press, Oxford, UK, 1998.

[137] Jeremy Goecks and Elizabeth Mynatt. Enabling privacy management in ubiquitous computing environments through trust and reputation systems. In *Proceedings of the 2002 ACM Conference on Computer Supported Cooperative Work (CSCW 2002, Workshop on Privacy in Digital Environments: Empowering Users)*, New Orleans, USA, November 2002. ACM Press. Available from: `www.cc.gatech.edu/fce/ecl/projects/saori/pubs/ReputationTrust-cscw2002.pdf`.

[138] Erving Goffman. *The Presentation of Self in Everyday Life*. Anchor-Doubleday, New York, USA, 1959.

[139] Dieter Gollmann. *Computer Security*. John Wiley & Sons, Chichester, England, 1999.

[140] Ken Gormley. One hundred years of privacy. *Wisconsin Law Review*, (1335), 1992. Available from: `cyber.law.harvard.edu/privacy/Gormley--100%20Years%20of%20Privacy.htm`.

[141] Elizabeth Gray, Paul O'Connell, Christian Jensen, Stefan Weber, Jean-Marc Seigneur, and Chen Yong. Towards a framework for assessing trust-based admission control in collaborative ad hoc applications. Technical Report 66, Department of Computer Science, Trinity College Dublin, 2002. Available from: `citeseer.nj.nec.com/gray02towards.html`.

[142] Elizabeth Gray, Jean-Marc Seigneur, Yong Chen, and Christian Jensen. Trust propagation in small worlds. In Paddy Nixon and Sotirios Terzis, editors, *Proceedings of the First International Conference on Trust Management (iTrust2003)*, volume 2692 of *Lecture Notes in Computer Science*, pages 239–254, Heraklion, Crete, Greece, May 2003. Springer-Verlag. Available from: `citeseer.nj.nec.com/575876.html`.

[143] John Gray, editor. *On Liberty and other Essays*. Oxford University Press, New York, USA, 1998.

[144] Jonathan Grudin. Desituating action: Digital representation of context. *Human-Computer Interaction*, 16(2-4):269–286, 2001.

[145] Marco Gruteser and Dirk Grunwald. Anonymous usage of location-based services through spatial and temporal cloaking. In *Proceedings of the First International Conference on Mobile Systems, Applications, and Services (MobiSys 2003)*, San Francisco, USA, May 2003. ACM USENIX. Available from: `systems.cs.colorado.edu/Papers/Generated/2003anonymousLbs.html`.

[146] Marco Gruteser and Xuan Liu. Protecting privacy in continuous location-tracking applications. *IEEE Security and Privacy*, 2(2):28–34, March/April 2004.

[147] Marco Gruteser, Graham Schelle, Ashish Jain, Rick Han, and Dirk Grunwald. Privacy-aware location sensor networks. In *Proceedings of HotOS IX: The 9th Workshop on Hot Topics in Operating Systems*, Lihue, Hawaii, USA, May 2003. USENIX. Available from: `www.usenix.org/events/hotos03/tech/full_papers/gruteser/gruteser.pdf`.

[148] The card up their sleeve. *The Guardian*, July 19, 2003. Available from: `www.guardian.co.uk/weekend/story/0,3605,999866,00.html`.

[149] Martin Gudgin, Marc Hadley, Jean-Jacques Moreau, and Henrik Frystyk Nielsen. SOAP version 1.2 part 1: Messaging framework. W3C Recommendation, World Wide Web Consortium (W3C), June 2003. Available from: `www.w3.org/TR/soap12-part1/`.

[150] Martin Gudgin, Marc Hadley, Jean-Jacques Moreau, and Henrik Frystyk Nielsen. SOAP version 1.2 part 2: Adjuncts. W3C Recommendation, World Wide Web Consortium (W3C), June 2003. Available from: `www.w3.org/TR/soap12-part2/`.

[151] John Hagel and Marc Singer. *Net Worth – Shaping Markets when Customers make the Rules*. Harvard Business School Press, Cambridge, USA, 1999.

[152] Mimi Hall and Barbara DeLollis. Plan to collect flier data canceled. *USA Today*, July 14, 2004. Available from: `www.usatoday.com/news/washington/2004-07-14-fly-plan_x.htm`.

[153] Chip soll Euro sicher machen. *Handelsblatt*, May 23, 2003. Available from: `www.handelsblatt.com/hbiwwwangebot/fn/relhbi/sfn/buildhbi/cn/GoArt!200104,201197,631871/SH/0/depot/0/index.html`.

[154] Holger Krull Hans-Werner Gellersen, Michael Beigl. The mediacup: Awareness technology embedded in an everyday object. In Hans-Werner Gellersen, editor, *Handheld & Ubiquitous Computing*, volume 1707 of *LNCS*, pages 308–310, Karlsruhe, Germany, 1999. TecO, University of Karlsruhe, Springer-Verlag. Available from: `www.teco.uni-karlsruhe.de/~michael/publication/mediacuphtml/`.

[155] Russel Hardin. *Do we want to trust in government?*, chapter 2, pages 22–41. In Warren [344], 1999.

[156] Harris Interactive. IBM multi-national consumer privacy survey, October 1999.

[157] Martin Hartmann. Einleitung. In Hartmann and Offe [158], pages 7–36.

[158] Martin Hartmann and Claus Offe, editors. *Vertrauen. Die Grundlage des sozialen Zusammenhalts*. Campus-Verlag, Frankfurt/Main, Germany, 2001.

[159] Christian Hauser and Matthias Kabatnik. Towards privacy support in a global location service. In *Proceedings of the IFIP Workship on IP and ATM Traffic Management*, pages 81–89, Paris, France, 2001. Available from: `www.ikr.uni-stuttgart.de/Content/Publications/Archive/Ha_EUNICE01_33986.pdf`.

[160] Mike Hazas and Andy Ward. A high performance privacy-oriented location system. In *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications (PerCom'03)*, pages 216–223, Fort Worth, USA, March 2003. IEEE Computer Society. Available from: `computer.org/proceedings/percom/1893/18930216abs.htm`.

[161] Urs Hengartner and Peter Steenkiste. Access control to information in pervasive computing environments. In *Proceedings of HotOS IX: The 9th Workshop on Hot Topics in Operating Systems*, Lihue, Hawaii, USA, May 2003. USENIX.

[162] Urs Hengartner and Peter Steenkiste. Protecting access to people location information. In Dieter Hutter, Günter Müller, Werner Stephan, and Markus Ullmann, editors, *Proceedings of the First International Conference on Security in Pervasive Computing (SPC 2003)*, volume 2802 of *LNCS*, March 12–14, 2003, Boppard, Germany, 2003. Springer-Verlag. Available from: `www-2.cs.cmu.edu/~uhengart/spc03.pdf`.

[163] Dirk Henrici and Paul Müller. Hash-based enhancement of location privacy for radio-frequency identification devices using varying identifiers. In Francis Lau and Hui Lei, editors, *Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops*, pages 149–153, Orlando, FL, USA, March 2004. IEEE Computer Society. Available from: `ieeexplore.ieee.org/xpl/tocresult.jsp?isNumber=28557&page=2`.

[164] Jeffrey Hightower and Gaetano Boriello. Location systems for ubiquitous computing. *IEEE Computer*, 33(8):57–66, August 2001.

[165] Lorenz Hilty, Siegfried Behrendt, Mathias Binswanger, Arend Bruinink, Lorenz Erdmann, Jürg Fröhlich, Andreas Köhler, Niels Kuster, Claudia Som, and Felix Würtenberger. Das Vorsorgeprinzip in der Informationsgesellschaft – Auswirkungen des Pervasive Computing auf Gesundheit und Umwelt. TA 46/2003, TA-Swiss, Berne, Switzerland, August 2003.

[166] Harry Hochheiser. The platform for privacy preference as a social protocol: An examination within the u.s. policy context.

ACM Transactions on Internet Technology (TOIT), 2(4):276–306, 2002.

[167] Jason I. Hong and James A. Landay. An architecture for privacy-sensitive ubiquitous computing. In *MobiSYS '04: Proceedings of the 2nd international conference on mobile systems, applications, and services*, pages 177–189. ACM Press, 2004.

[168] US House of Representatives. The transportation security administration's perspective on aviation security. Memo to the Committee on Transportation and Infrastructure, October 2003. Available from: `www.house.gov/transportation/aviation/10-16-03/10-16-03memo.html`.

[169] Xiaorui R. Hu, Zhangxi X. Lin, and Han Zhang. Myth or reality: effect of trustpromoting seals in electronic markets. In *Proceedings of the Eleventh Annual Workshop on Information Technologies and Systems (WITS)*, pages 65–70, New Orleans, USA, 2001.

[170] Richard Hull, Bharat Kumar, Daniel Lieuwen, Peter F. Patel-Schneider, Arnaud Sahuguet, Sriram Varadarajan, and Avinash Vyas. Enabling context-aware and privacy-conscious user data sharing. In *Proceedings of the 2004 IEEE International Conference on Mobile Data Management (MDM 2004)*, pages 187–198, January 2004. Available from: `ieeexplore.ieee.org/xpl/tocresult.jsp?isNumber=28243&page=1`.

[171] Hilary Hutchinson, Wendy Mackay, Bo Westerlund, Benjamin B. Bederson, Allison Druin, Catherine Plaisant, Michel Beaudouin-Lafon, Stéphane Conversy, Helen Evans, Heiko Hansen, Nicolas Roussel, and Björn Eiderbäck. Technology probes: inspiring design for and with families. In *Proceedings of the SIGCHI conference on Human factors in computing systems (CHI '03)*, pages 17–24, Ft. Lauderdale, USA, 2003. ACM Press.

[172] IBM Global Services. Transforming the appliance industy: Revenue streams through service. White paper, 2001. Available from: `www.ibm.com/pvc/tech/pdf/gsoee203.pdf`.

[173] IEEE – Institute of Electrical and Electronics Engineers. Part 15.4: Wireless medium access control (MAC) and physical layer (PHY) specifications for low-rate wireless personal area networks

(LR-WPANs). IEEE Standard for information technology – telecommunications and information exchange between systems – local and metropolitan area networks – specific requirements. 802.15.4, IEEE Computer Society, New York, USA, October 2003. Available from: standards.ieee.org/getieee802/download/802.15.4-2003.pdf.

[174] UK Information Commissioner. The Anti-Terrorism, Crime and Security Act 2001: Retention and disclosure of communications data – summary of counsels' advice, 2002. Available from: www.privacyinternational.org/countries/uk/surveillance/ic-terror-opinion.htm.

[175] Ronald Inglehart. *Trust, well-being and democracy*, chapter 4, pages 88–120. In Warren [344], 1999.

[176] Sozo Inoue and Hiroto Yasuura. RFID privacy using user-controllable uniqueness. MIT RFID Privacy Workshop, Cambridge, MA, USA, November 15, 2004. Available from: www.rfidprivacy.org/papers/sozo_inoue.pdf.

[177] ISO – International Organisation for Standardization. 7498-2 basic reference model for open systems interconnection (osi) part 2: Security architecture. ISO/IEC 7498-2, Geneva, Switzerland, 1988. As cited in [139].

[178] ISO – International Organisation for Standardization. Evaluation criteria for IT security – Part 2: Security functional requirements, 1999. Available from: www.clusit.it/whitepapers/iso15408-2.pdf.

[179] ISO – International Organization for Standardization. 18000 Information technology automatic identification and data capture techniques - Radio frequency identification for item management air interface. ISO/IEC 18000, Geneva, Switzerland, 2003.

[180] Uwe Jendricke and Daniela Gerd tom Markotten. Usability meets security – the identity-manager as your personal security assistant for the internet. In *Proceedings of the 16th Annual Computer Security Applications Conference*, pages 344–353, December 2000. Available from: tserv.iig.uni-freiburg.de/telematik/forschung/projekte/kom_technik/atus/publications/JeGe2000.pdf.

[181] Uwe Jendricke, Michael Kreutzer, and Alf Zugenmeier. Mobile identity management. Security Workshop at Ubicomp 2002, September 2002. Available from: `www.inf.ethz.ch/vs/publ/se/identity-management.pdf`.

[182] Debora G. Johnson. *Ethical Issues in Engineering*. Prentice Hall, Englewood Cliffs, NJ, USA, 1991.

[183] Debora G. Johnson. Sorting out the uniqueness of computer-ethical issues. *Etica & Politica*, 1(2), 1999. Excerpt from the 3rd edition of *Computer Ethics*, Prentice Hall, 2001. Available from: `www.units.it/~etica/1999_2/johnson.html`.

[184] Audun Jøsang. The right type of trust for distributed systems. In Cathy Meadows, editor, *Proceedings of the 1996 Workshop on New Security Paradigms*, Lake Arrowhead, CA, USA, May 1997. ACM Press. Available from: `citeseer.nj.nec.com/josang96right.html`.

[185] Ari Juels and Ravikanth Pappu. Squealing Euros: Privacy protection in RFID-enabled banknotes. In Rebecca N. Wright, editor, *Proceedings of the Seventh International Conference on Financial Cryptography (FC 2003)*, volume 2742 of *LNCS*, pages 103–121, Guadeloupe, French West Indies, January 2003. Springer-Verlag. Available from: `www.rsasecurity.com/rsalabs/staff/bios/ajuels/publications/squealing-euros/SquealingEuros.pdf`.

[186] Ari Juels, Ronald L. Rivest, and Michael Szydlo. The blocker tag: Selective blocking of RFID tags for consumer privacy. In Sushil Jajodia, Vijay Atluri, and Trent Jaeger, editors, *Proceedings of the 10th ACM Conference on Computer and Communication Security*, pages 103–111, Washington, D.C., USA, 2003. ACM Press. Available from: `portal.acm.org/citation.cfm?id=948126&coll=Portal`.

[187] Lalana Kagal, Tim Finin, and Anupam Joshi. Trust-based security in pervasive computing environments. *IEEE Computer*, 34(12):154–157, December 2001.

[188] Joseph M. Kahn, Randy H. Katz, and Kristofer S. J. Pister. Next century challenges: Mobile networking for "smart dust". In *Proceedings of the ACM/IEEE International Conference on Mobile*

Computing and Networking (MobiCom 99), pages 271–278, Seattle, WA, USA, August 1999.

[189] Immanuel Kant. *Critique of Practical Reason*. Longmans, Green, and Co., London, UK, fifth edition, 1898. Available from: `oll.libertyfund.org/ToC/0212.php`.

[190] Günter Karjoth and Matthias Schunter. A privacy policy model for enterprises. In *Proceedings of the 15th IEEE Computer Security Foundations Workshop (CSFW-15)*, pages 271–281, Cape Breton, Nova Scotia, Canada, June 2002. Available from: `ieeexplore.ieee.org/xpl/tocresult.jsp?isNumber=21985&page=1`.

[191] Günter Karjoth, Matthias Schunter, Els Van Herreweghen, and Michael Waidner. Amending P3P for clearer privacy promises. In *Proceedings of the 14th International Workshop on Database and Expert Systems Applications (DEXA'03)*, pages 445–449, Prague, Czech Republic, September 2003. IEEE Press. Available from: `ieeexplore.ieee.org/xpl/tocresult.jsp?isNumber=27592&page=4`.

[192] Günter Karjoth, Matthias Schunter, and Michael Waidner. Privacy-enabled services for enterprises. In *Proceedings of the 13th International Workshop on Database and Expert Systems Applications (DEXA'02)*, pages 405–409, Aix-en-Provence, France, September 2002. IEEE Press. Available from: `ieeexplore.ieee.org/xpl/tocresult.jsp?isNumber=22410&page=5`.

[193] Ichiro Kawachi, Bruce P. Kennedy, Kimberly Lochner, and Deborah Prothrow-Smith. Social capital, income inequality, and mortality. *American Journal of Public Health*, 87(9):1491–1498, 1997. As cited in [277].

[194] Dogan Kesdogan, Peter Reichl, and Klaus Junghärtchen. Distributed temporary pseudonyms: A new approach for protecting location information in mobile communication networks. In Jean-Jacques Quisquater, Yves Deswarte, Catherine Meadows, and Dieter Gollmann, editors, *Computer Security – Proceedings of the Fifth European Symposium on Research in Computer Se-*

curity (ESORICS '98), number 1485 in LNCS, pages 295–312, Louvain-la-Neuve, Belgium, 1998. Springer-Verlag.

[195] Andreas Kirsch. RFIDs in euro banknotes. RFIDwatch Website, May 23, 2003. Available from: `www.unwatched.org/article4.html`.

[196] Alfred Kobsa and Jörg Schreck. Privacy through pseudonymity in user-adaptive systems. *ACM Transactions on Internet Technology*, 3(2):149–183, 2003.

[197] Gerd Kortuem, Zary Segall, and Thaddeus G. Cowan Thompson. Close encounters: Supporting mobile collaboration through interchange of user profiles. In Gellersen [134], pages 171–185. Available from: `citeseer.nj.nec.com/kortuem99close.html`.

[198] Stacy E. Kovar, Kimberly G. Burke, and Brian R. Kovar. Consumer responses to the CPA WEBTRUST assurance. *Journal of Information Systems*, 14(1):17–35, 2000.

[199] Dennis Kügler. Risiko Reisepass. *c't*, (3):84–89, February 2005.

[200] Rainer Kuhlen. *Informationsethik*. UTB. Universitätsverlag Konstanz, Konstanz, Germany, 2004. Available from: `www.inf-wiss.uni-konstanz.de/CURR/summer04/infethik04/zeit-arbeitsplan-ie-kn04.html`.

[201] Steven Kuhn. Prisoner's Dilemma. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Fall 2003. Available from: `plato.stanford.edu/archives/fall2003/entries/prisoner-dilemma/`.

[202] Olli Lagerspetz. Vertrauen als geistiges Phänomen. In Hartmann and Offe [158], chapter 1, pages 85–113.

[203] Christel Lane and Reinhard Bachmann. The social constitution of trust: supplier relations in britain and germany. *Organization Studies*, 17(3):365–395, 1996. Available from: `www.findarticles.com/p/articles/mi_m4339/is_n3_v17/ai_18735887`.

[204] Marc Langheinrich. Neuer Standard für Online-Datenschutz. *digma: Zeitschrift für Datenrecht und Informationssicherheit*, 1(1):32–34, 2001.

[205] Marc Langheinrich. Privacy by design – principles of privacy-aware ubiquitous systems. In Abowd et al. [2], pages 273–291. Available from: www.vs.inf.ethz.ch/publ/papers/privacy-principles.pdf.

[206] Marc Langheinrich. A privacy awareness system for ubiquitous computing environments. In Gaetano Borriello and Lars Erik Holmquist, editors, *Proceedings of the Fourth International Conference on Ubiquitous Computing (UbiComp 2002)*, number 2498 in LNCS, pages 237–245, Gothenburg, Sweden, September 2002. Springer-Verlag.

[207] Marc Langheinrich. The dc-privacy troubadour – assessing privacy implications of dc-projects. Designing for Privacy Workshop, DC Tales Conference, June 2003. Available from: www.vs.inf.ethz.ch/publ/papers/dctales-privacy.pdf.

[208] Marc Langheinrich. When trust does not compute – the role of trust in ubiquitous computing. Workshop on Privacy at UbiComp 2003, October 2003.

[209] Marc Langheinrich. Die Privatsphäre im Ubiquitous Computing – Datenschutzaspekte der RFID-Technologie. In Elgar Fleisch and Friedemann Mattern, editors, *Das Internet der Dinge – Ubiquitous Computing und RFID in der Praxis*. Springer-Verlag, 2005.

[210] Marc Langheinrich and Friedemann Mattern. Digitalisierung des Alltags. Was ist Pervasive Computing? *Aus Politik und Zeitgeschichte*, (42):6–12, October 2003.

[211] Kenneth C. Laudon. Extensions to the theory of markets and privacy: Mechanics of pricing information. In William N. Daley and Larry Irving, editors, *Privacy and Self-Regulation in the Information Age*, national telecommunications and information administration report 1D. U.S. Department of Commerce, Washington, USA, 1997. As cited in [302]. Available from: www.ntia.doc.gov/reports/privacy/selfreg1.htm#1D.

[212] Cédric Laurant, editor. *Privacy and Human Rights 2003*. EPIC and Privacy International, London, UK, 2003. Available from: www.privacyinternational.org/survey/phr2003/.

[213] Ching Law, Kayi Lee, and Kai-Yeung Siu. Efficient memoryless protocol for tag identification (extended abstract). In *Proceedings of the Fourth International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, pages 75–84. ACM Press, 2000. Available from: `portal.acm.org/citation.cfm?id=345865&dl=ACM&coll=portal`.

[214] Scott Lederer, Anind K. Dey, and Jennifer Mankoff. A conceptual model and a metaphor of everyday privacy in ubiquitous computing. Technical Report UCB-CSD-02-1188, University of California at Berkeley, July 2002. Available from: `www.cs.berkeley.edu/~lederer/research/privacy/everydayprivacy.pdf`.

[215] Scott Lederer, Jason I. Hong, Anind K. Dey, and James A. Landay. Personal privacy through understanding and action: Five pitfalls for designers. *Personal and Ubiquitous Computing*, 8(6):440–454, September 2004.

[216] Scott Lederer, Jason I. Hong, Xiaodong Jiang, Anind K. Dey, James A. Landay, and Jennifer Mankoff. Towards everyday privacy for ubiquitous computing. Technical Report UCB-CSD-03-1283, University of California at Berkeley, October 2003. Available from: `www.cs.berkeley.edu/projects/io/publications/privacy-techreport03a.pdf`.

[217] Lawrence Lessig. *Code and Other Laws of Cyberspace*. Basic Books, New York, USA, 1999.

[218] Lawrence Lessig. Privacy as property. *Social Research*, 69(1):247–269, 2002. Available from: `www.findarticles.com/p/articles/mi_m2267/is_1_69/ai_88584150`.

[219] John Lettice. EU biometric RFID scheme unworkable, says EU tech report. *The Register*, December 23, 2004. Available from: `www.theregister.co.uk/2004/12/23/eu_rfid_visa_trashes_self/`.

[220] John Lettice. Plugs to be pulled on EU biometric visa scheme? *The Register*, January 7, 2005. Available from: `www.theregister.co.uk/2005/01/07/cards_for_bio_visa/`.

[221] Roy J. Lewicki and Barbara Benedict Bunker. Developing and maintaining trust in working relationships. In Roderick M.

Kramer and Tom R. Tyler, editors, *Trust in Organizations: Frontiers of Theory and Research*, pages 114–139. Sage Publications, Thousand Oaks, CA, USA, 1996.

[222] John Leyden. FBI apology for Madrid bomb fingerprint fiasco. *The Register*, May 26, 2004. Available from: `www.theregister.co.uk/2004/05/26/fbi_madrid_blunder/`.

[223] Dahlia Lithwick and Julia Turner. A guide to the patriot act. *MSN Slate Magazine*, September 8, 2003. Available from: `slate.msn.com/id/2087984/`.

[224] Niklas Luhmann. Familiarity, confidence, trust: Problems and alternatives. In Gambetta [129], pages 94–107. Available from: `www.sociology.ox.ac.uk/papers/trustbook.html`.

[225] Niklas Luhmann. *Vertrauen: ein Mechanismus der Reduktion sozialer Komplexität*. Lucius und Lucius, Stuttgart, Germany, 4th edition edition, 2000.

[226] Niklas Luhmann. Vertrautheit, Zuversicht, Vertrauen: Probleme und Alternativen. In Hartmann and Offe [158], chapter 2, pages 143–160.

[227] David Lyon. Terrorism and surveillance: Security, freedom, and justice after september 11 2001. Privacy Lecture Series, November 12, 2001. Available from: `privacy.openflows.org/pdf/lyon_paper.pdf`.

[228] David Lyon, editor. *Surveillance as Social Sorting: Privacy, Risk and Automated Discrimination*. Routledge, 2002.

[229] Steve Mann, Jason Nolan, and Barry Wellman. Sousveillance: Inventing and using wearable computing devices for data collection in surveillance environments. *Surveillance & Society*, 1(3):331–355, July 2003. Available from: `www.surveillance-and-society.org/journalv1i3.htm`.

[230] Jane Mansbridge. *Altruistic Trust*, chapter 10, pages 290–309. In Warren [344], 1999.

[231] Harvey C. Mansfield and Delba Winthrop, editors. *Alexis de Tocqueville: Democracy in America*. University of Chicago Press, Chicago, USA, 2000.

[232] Stephen Paul Marsh. *Formalising Trust as a Computational Concept*. PhD thesis, Department of Computer Science and Mathematics, University of Sterling, April 1994.

[233] Gary T. Marx. Murky conceptual waters: The public and the private. *Ethics and Information Technology*, 3(3):157–169, 2001. Available from: `web.mit.edu/gtmarx/www/murkypublicandprivate.html`.

[234] Friedemann Mattern. The vision and technical foundations of ubiquitous computing. *Upgrade*, 2(5):2–6, October 2001.

[235] Victor Mayer-Schönberger. Generational development of data protection in Europe. In Agre and Rotenberg [14], chapter 8, pages 219–242.

[236] D. Harrison McKnight and Norman L. Chervany. The meanings of trust. Working Paper 96-04, Management Information Systems Research Center, Carlson School of Management, University of Minnesota, 1996. Last revised: April 1, 2000. Available from: `misrc.umn.edu/workingpapers/fullPapers/1996/9604_040100.pdf`.

[237] D. Harrison McKnight, Larry L. Cummings, and Norman L. Chervany. Trust formation in new organizational relationships. Working Paper 96-01, Management Information Systems Research Center, Carlson School of Management, University of Minnesota, 1996. Last revised: August 1, 1997. Available from: `misrc.umn.edu/workingpapers/fullPapers/1996/9601_080197.pdf`.

[238] Colin H.H. McNairn and Alexander K. Scott. *Privacy Law in Canada*. Butterworths, Markham, Canada, 2001. As discussed in [248].

[239] Michael Mealling. *Auto-ID Object Name Service (ONS) 1.0*, August 2003. Available from: `www.epcglobalinc.org/standards_technology/Secure/v1.0/WD-ons-1.0-20030930.pdf`.

[240] Anti-spam law goes into force in Europe. *Mercury News*, October 31, 2003. Available from: `www.siliconvalley.com/mld/siliconvalley/7151271.htm`.

[241] John Stuart Mill. On liberty. In Gray [143], chapter 1, pages 5–130.

[242] John Stuart Mill. Utilitarianism. In Gray [143], chapter 2, pages 131–204.

[243] Michael J. Miller. E-relationships are vital. *PC Magazine*, 18(20):4, November 16 1999. As cited in [93].

[244] Paul Miotti. Conception and implementation of a privacy-supportive database. Master's thesis, ETH Zurich, Zurich, Switzerland, 2002.

[245] Nilo Mitra. SOAP version 1.2 part 0: Primer. W3C Recommendation, World Wide Web Consortium (W3C), June 2003. Available from: `www.w3.org/TR/soap12-part0/`.

[246] Gordon Moore. Cramming more components onto integrated circuits. *Electronics*, (38):114–117, April 1965.

[247] James H. Moore. What is computer ethics? *Metaphilosophy*, 16(4):266–275, October 1985. As cited in [183].

[248] Anne Mussett. Book review: Privacy law in canada. *Canadian Journal of Law and Technology*, 1(2), July 2002. Available from: `cjlt.dal.ca/vol1_no2/pdfarticles/mussett.pdf`.

[249] Ginger Myles, Adrian Friday, and Nigel Davies. Preserving privacy in environments with location-based applications. *IEEE Pervasive Computing*, 2(1):56–64, 2003.

[250] Elizabeth D. Mynatt, Jim Rowan, Sarah Craighill, and Annie Jacobs. Digital family portraits: supporting peace of mind for extended family members. In *Proceedings of the SIGCHI conference on Human factors in computing systems (CHI '01)*, pages 333–340. ACM Press, 2001.

[251] Kris Nagel, Cory D. Kidd, Thomas O'Connell, Anind K. Dey, and Gregory D. Abowd. The family intercom: Developing a context-aware audio communication system. In Abowd et al. [2], pages 176–183.

[252] Carmen Neustaedter and Saul Greenberg. The design of a context-aware home media space for balancing privacy and awareness. In Anind K. Day, Albrecht Schmidt, and Joseph F. McCarthy, editors, *Proceedings of the 5th International Conference on Ubiquitous Computing (Ubicomp 2003)*, pages 297–314, Seattle, USA, 2003. Springer-Verlag.

[253] David H. Nguyen and Elizabeth D. Mynatt. Privacy mirrors: Understanding and shaping socio-technical ubiquitous computing systems. Technical Report GIT-GVU-02-16, Georgia Institute of Technology, Atlanta, GA, USA, 2002. Available from: `erstwhile.org/~dnguyen/writings/PrivacyMirrors.pdf`.

[254] Andrew M. Odlyzko. Privacy, economics, and price discrimination on the internet. *Proceedings of the Fifth International Conference on Electronic Commerce (ICEC2003)*, pages 355–366, 2003. Also appears as chapter 15 in [55]. Available from: `www.dtc.umn.edu/~odlyzko/doc/privacy.economics.pdf`.

[255] Claus Offe. Wie können wir unseren Mitbürgern vertrauen? In Hartmann and Offe [158], chapter 3, pages 241–294.

[256] Robert O'Harrow, Jr. U.S. backs Florida's new counterterrorism database. *The Washington Post*, page A01, August 6, 2003. Available from: `www.washingtonpost.com/ac2/wp-dyn/A21872-2003Aug5`.

[257] Miyako Ohkubo, Koutarou Suzuki, and Shingo Kinoshita. Cryptographic approach to "privacy-friendly" tags. MIT RFID Privacy Workshop, Cambridge, MA, USA, November 15, 2004. Available from: `www.rfidprivacy.org/papers/ohkubo.pdf`.

[258] Onora O'Neill. *A Question of Trust – The BBC Reith Lectures 2002*. Cambridge University Press, Cambridge, UK, 2002.

[259] Tim O'Reilly. The fuss about Gmail and privacy: Nine reasons why it's bogus. O'Reilly Developer Weblogs, April 16, 2004. Available from: `www.oreillynet.com/pub/wlg/4707`.

[260] Organisation for Economic Co-operation and Development (OECD). Recommendation of the council concerning guidelines

governing the protection of privacy and transborder flows of personal data, September 1980. Available from: www.privacy.gov.au/publications/oecdgls.pdf.

[261] Elizabeth Paton-Simpson. Privacy and the reasonable paranoid: The protection of privacy in public places. *University of Toronto Law Journal*, 50(3), 2000. Available from: www.utpjournals.com/product/utlj/503/503_paton.html.

[262] Andrew S. Patrick. Building trustworthy software agents. *IEEE Internet Computing, Special Issue on the Technology of Trust*, 6(6):46–53, November 2002. Available from: www.computer.org/internet/ic2002/w6toc.htm.

[263] Andreas Pfitzmann and Marit Koehntopp. Anonymity, unobservability, and pseudonymity – a proposal for terminology. In Hannes Federrath, editor, *Proceedings Workshop on Design Issues in Anonymity and Unobservability*, volume 2009 of *LNCS*. Springer-Verlag, 2001. Available from: marit.koehntopp.de/pub/anon/Anon_Terminology.pdf.

[264] Joseph P. Pickett, editor. *The American Heritage College Dictionary*. Houghton Mifflin Co, 4th edition, April 2002.

[265] Suzy Platt, editor. *Respectfully Quoted: A Dictionary of Quotations Requested from the Congressional Research Service*. Library of Congress, Washington, D.C., USA, 1989. Available from: www.bartleby.com/73/.

[266] Stephen G. Post, editor. *Encyclopedia of Bioethics*. MacMillan Reference, 3rd edition, 2003.

[267] Washington Post, Kaiser Family Foundation, and Harvard University Survey Project. Why don't Americans trust the government?, 1996. Available from: www.kff.org/kaiserpolls/1110-governs.cfm.

[268] Kevin Poulsen. EU goes on biometric LSD trip. *The Register*, February 3, 2005. Available from: www.theregister.co.uk/2005/02/03/biometric_lsd_trip/.

[269] Privacilla.org. The privacy torts: How U.S. state law quietly leads the way in privacy protection. Privacilla.org Website,

2002. Available from: www.privacilla.org/releases/Torts_Report.html.

[270] Privacy Rights Clearinghouse. A review of the fair information principles: The foundation of privacy public policy, February 2004. Available from: www.privacyrights.org/ar/fairinfo.htm.

[271] Privacy Rights Clearinghouse. Thirty-one privacy and civil liberties organizations urge google to suspend Gmail. Press Release, April 6, 2004. Available from: www.privacyrights.org/ar/GmailLetter.htm.

[272] Nissanka B. Priyantha, Anit Chakraborty, and Hari Balakrishnan. The cricket location-support system. In *Proc. of the Sixth Annual ACM International Conference on Mobile Computing and Networking (MOBICOM 2000)*, pages 32–43, Boston, USA, 2000. ACM Press. Available from: nms.lcs.mit.edu/papers/cricket.pdf.

[273] William Prosser. Privacy. *California Law Journal*, 48:383–423, 1960. As cited in [322].

[274] Robert D. Putnam. Bowling alone: America's declining social capital. *Journal of Democracy*, 6(1):65–78, January 1995.

[275] Robert D. Putnam. *Bowling Alone: The Collapse and Revival of American Community*. Simon and Schuster, New York, USA, 2000.

[276] Robert D. Putnam. Bowling together. *The American Prospect*, 13(3), February 2002. Available from: www.prospect.org/print/V13/3/putnam-r.html.

[277] Wendy M. Rahn, Kwang Suk Yoon, Michael Garet, Steven Lopson, and Katherine Loflin. Geographies of trust: Explaining intercommunity variation in general social trust using hierarchical linear modeling. In *58th Annual Conference of the American Association for Public Opinion Research*, Nashville, TN, USA, May 16, 2003.

[278] Cliff Randell and Henk L. Muller. Low cost indoor positioning system. In Abowd et al. [2], pages 42–48. Available

from: link.springer.de/link/service/series/0558/bibs/2201/22010042.htm.

[279] John Rawls. *A Theory of Justice*. Harvard University Press, Cambridge, MA, USA, revised edition, 1999.

[280] John Rawls. *Lectures on the History of Moral Philosophy*. Harvard University Press, Cambridge, MA, USA, 2000.

[281] Joseph Reagle. A P3Passurance signature profile. W3C Note, World Wide Web Consortium, February 2001. Available from: www.w3.org/TR/xmldsig-p3p-profile/.

[282] Tony Redriguez. Plan the project or pay the price. *DM Review*, February 2002. Available from: www.dmreview.com/editorial/newsletter_article.cfm?nl=bireport&articleId=4625&issue=634.

[283] Drummond Reed and Geoffrey Strongin. The Dataweb: An introduction to XDI. White paper, OASIS XDI Technical Committee, April 2004. Available from: www.oasis-open.org/committees/download.php/6434/wd-xdi-intro-white-paper-2004-04-12.pdf.

[284] Joel R. Reidenberg. E-commerce and trans-atlantic privacy. *Houston Law Review*, 38(3):717–749, September 2001. Available from: reidenberg.home.sprynet.com/Transatlantic_Privacy.pdf.

[285] Michael Reissenberger. 50 Jahre Bundesverfassungsgericht: Volkszählung. DeutschlandRadio Schwerpunktthema, January 4, 2004. Available from: www.dradio.de/homepage/schwerpunkt-verfassungsgericht-010904.html.

[286] Michael K. Reiter and Aviel D. Rubin. Anonymous web transactions with crowds. *Communications of the ACM*, 42(2):32–48, 1999.

[287] Paul Resnick and James Miller. PICS: Internet access controls without censorship. *Communications of the ACM*, 39(10):87–93, 1996. Available from: www.w3.org/PICS/iacwcv2.htm.

[288] Hitachi unveils integrated RFID tag. *RFID Journal*, September 4, 2003. Available from: `www.rfidjournal.com/article/articleview/556/1/1/`.

[289] Bradley Rhodes. The wearable remembrance agent: A system for augmented memory. *Personal Technologies Journal. Special Issue on Wearable Computing*, 1:218–224, 1997.

[290] Bradley Rhodes, Nelson Minar, and Josh Weaver. Wearable computing meets ubiquitous computing - reaping the best of both worlds. In *Proc. of The Third International Symposium on Wearable Computers (ISWC '99)*, pages 141–149, San Francisco, CA, October 1999. Available from: `rhodes.www.media.mit.edu/people/rhodes/Papers/wearhive.html`.

[291] Jens Riegelsberger, M. Angela Sasse, and John D. McCarthy. Shiny happy people building trust? Photos on e-commerce Websites and consumer trust. In Gilbert Cockton and Panu Korhonen, editors, *Proceedings of the 2003 Conference on Human Factors in Computing Systems (CHI'03)*, pages 121–128, Ft. Lauderdale, FL, USA, April 2003. ACM Press.

[292] Ron Rivest. The MD5 message-digest algorithm. Request for Comments 1321, MIT Laboratory for Computer Science and RSA Data Security, Inc., 1994. Available from: `www.cse.ohio-state.edu/cs/Services/rfc/rfc-text/rfc1321.txt`.

[293] Ron L. Rivest. Whither information security? SCS Distinguished Lecture, March 15, 2001. Available from: `wean1.ulib.org/Lectures/Distinguished%20Lectures/2001/03.0%20Ronald%20L%20Rivest/6SLIDES/security.ppt`.

[294] Ron L. Rivest, Adi Shamir, and Leonard M. Adelman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[295] Tom Rodden, Adrian Friday, Henk Muller, and Alan Dix. A lightweight approach to managing privacy in location-based services. Technical Report Equator-02-058, University of Nottingham and Lancaster University and University of Bristol, 2002. Available from: `www.equator.ac.uk/PublicationStore/2002-rodden-1.pdf`.

[296] Jothy Rosenberg and David Remy. *Securing Web Services Security with WS-Security*. Sams Publishing, 2004.

[297] Richard Rosenfeld, Steven F. Messner, and Eric P. Baumer. Social captial and homicide. *Social Forces*, 80(1):283–309, 2001. As cited in [277].

[298] Beate Rössler. *Der Wert des Privaten*. Suhrkamp Verlag, Frankfurt/Main, Germany, 2001.

[299] Alexander Roßnagel. Regulierung und Selbstregulierung im Datenschutz. In Herbert Kubicek, Hans-Joachim Braczyk, Dieter Klumpp, and Alexander Roßnagel, editors, *Global@home – Informations- und Dienstleistungsstrukturen der Zukunft*, volume 8 of *Jahrbuch Telekommunikation und Gesellschaft*, pages 385–391. Hüthig-Verlag, 2000.

[300] Alexander Roßnagel, Andreas Pfitzmann, and Hansjürgen Garstka. *Modernisierung des Datenschutzrechts*. Bundesministerium des Inneren, Berlin, Germany, 2001. Available from: `www.bmi.bund.de/downloadde/11659/Download.pdf`.

[301] Nicolas Roussel, Helen Evans, and Heiko Hansen. Proximity as an interface for video communication. *IEEE MultiMedia*, 11(3):12–16, 2004. Available from: `www.lri.fr/~roussel/publications/ieee-multimedia-04.pdf`.

[302] Pamela Samuelson. Privacy as intellectual property? *Stanford Law Review*, 52:1125–1173, 2000. Available from: `www.sims.berkeley.edu/~pam/papers/privasip_draft.pdf`.

[303] Sanjay E. Sarma, Stephen A. Weis, and Daniel W. Engels. RFID systems and security and privacy implications. In Burton S. Jr. Kaliski, Cetin K. Koc, and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems (CHES 2002) – Proceedings of the Fourth International Workshop*, volume 2523 of *LNCS*, pages 454–469, Redwood Shores, CA, USA, 2002. Springer-Verlag.

[304] Sidney C. Schaer. Suiting up for life underwater – Predictions from the past that haven't come true... yet. Our Future – Special report on Newsday.com. Available from: `future.newsday.com/4/fbak0416.htm`.

[305] Bruce Schimmel. The Bork shield. *Philadelphia City Paper*, April 11-17 2002. Available from: `www.citypaper.net/articles/2002-04-11/canon.shtml`.

[306] Albrecht Schmidt, Kofi Asante Aidoo, Antti Takaluomai, Urpo Tuomelai, Kristof Van Laerhoven, and Walter Van de Velde. Advanced interaction in context. In Gellersen [134], pages 89–101. Available from: `www.comp.lancs.ac.uk/~albrecht/pubs/pdf/schmidt_huc99_advanced_interaction_context.pdf`.

[307] Jay Schneider, Gerd Kortuem, Joe Jager, Steve Fickas, and Zary Segall. Disseminating trust information in wearable communities. In *Proceedings of the Second International Symposium on Handheld and Ubiquitous Computing (HUC2K)*, Bristol, UK, September 2000. Available from: `www.comp.lancs.ac.uk/~kortuem/publications/HUC2K.pdf`.

[308] Roland Schneider. RFID privacy platform. Master's thesis, ETH Zurich, Zurich, Switzerland, 2004.

[309] Bruce Schneier. *Applied Cryptography*. John Wiley & Sons, New York, 1996.

[310] Bruce Schneier. RFID passports. Schneier on Security Weblog, October 4, 2004. Available from: `www.schneier.com/blog/archives/2004/10/rfid_passports.html`.

[311] Henning Schulzrinne, John B. Morris, Hannes Tschofening, Jorge R. Cuellar, James Polk, and Jonathan Rosenberg. A document format for expressing privacy preferences. Internet-draft, IETF, 2004. Available from: `www.ietf.org/internet-drafts/draft-ietf-geopriv-common-policy-03.txt`.

[312] Dietrich Schwanitz. *Bildung – Alles was man wissen muss*. Goldmann Verlag, Munich, Germany, 2002.

[313] Peter Schwartz and Peter Leyden. The long boom: A history of the future, 1980 - 2020. *Wired Magazine*, 5(7), July 1997. Available from: `www.wired.com/wired/archive/5.07/longboom.html`.

[314] John Scott. Rational choice theory. In Gary Browning, Abigail Halcli, and Frank Webster, editors, *Understanding Contemporary Society: Theories of the Present*. Sage Publication, London,

UK, February 2000. Available from: privatewww.essex.ac.uk/~scottj/socscot7.htm.

[315] Andrew Senior, Sharath Pankanti, Arun Hampapur, Lisa Brown, Ying-Li Tian, and Ahmet Ekin. Blinkering surveillance: Enabling video privacy through computer vision. Technical Report RC22886, IBM, Thomas J. Watson Research Center, Yorktown Hights, USA, 2003. Available from: www.research.ibm.com/peoplevision/rc22886.pdf.

[316] Brian Shand, Nathan Dimmock, and Jean Bacon. Trust for ubiquitous, transparent collaboration. In *Proceedings of the First Annual IEEE Conference on Pervasive Computing and Communications (PerCom 2003)*, pages 153–160, Dallas-Ft. Worth, TX, USA, March 2003. IEEE-Press. Available from: citeseer.ist.psu.edu/shand03trust.html.

[317] Narendar Shankar and William A. Arbaugh. On trust for ubiquitous computing. Workshop on Security in Ubiquitous Computing at UbiComp 2002, October 2002. Available from: www.teco.edu/~philip/ubicomp2002ws/.

[318] Jayavel Shanmugasundaram, Eugene J. Shekita, Jerry Kiernan, Rajasekar Krishnamurthy, Stratis Viglas, Jeffrey F. Naughton, and Igor Tatarinov. A general techniques for querying XML documents using a relational database system. *SIGMOD Record*, 30(3):20–26, 2001.

[319] Arnold Simmel. Privacy. In *International Encyclopedia of the Social Sciences*, page 480. MacMillan, 1968. As cited in [57].

[320] Peter Singer. Ethics. In *Encyclopædia Britannica*. Deluxe Edition CD-ROM, 2004.

[321] Robert Ellis Smith. The law of privacy in a nutshell. *Privacy Journal*, 19(6):50–51, 1993.

[322] Daniel J. Solove and Marc Rotenberg. *Information Privacy Law*. Aspen Publishers, New York, USA, 2003.

[323] Innere Sicherheit: Totes Pferd. *Der Spiegel*, (11):48, March 2004. Available from: www.spiegel.de/spiegel/inhalt/0,1518,ausg-1395,00.html.

[324] Mike Spreitzer and Marvin Theimer. Providing location information in a ubiquitous computing environment (panel session). In *Proceedings of the Fourteenth ACM Symposium on Operating Systems Principles (SOSP '93)*, pages 270–283. ACM Press, 1993.

[325] Mark Staeheli. Konzeption und Implementation eines Privacy Proxies. Master's thesis, ETH Zurich, Zurich, Switzerland, 2001.

[326] Frank Stajano. *Security for ubiquitous computing*. John Wiley & Sons, Ltd, 2002.

[327] Ross Stapleton-Gray. Scanning the horizon: A skeptical view of RFIDs on the shelves. MIT RFID Privacy Workshop, Cambridge, MA, USA, November 15, 2004. Available from: `www.rfidprivacy.org/papers/stapleton-gray3.pdf`.

[328] Mark Stefik. Trusted systems. *Scientific American*, 276(3):68–71, March 1997. Available from: `www.hackvan.com/pub/stig/articles/trusted-systems/0397stefik.html`.

[329] Mark Stefik, editor. *The Internet Edge: Social, Technical, and Legal Challenges for a Networked World*. MIT Press, Cambridge, USA, 1999.

[330] Latanya Sweeney. Achieving k-anonymity privacy protection using generalization and suppression. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, 10(5):571–588, 2002. Available from: `privacy.cs.cmu.edu/people/sweeney/kanonymity2.pdf`.

[331] Latanya Sweeney. k-anonymity: a model for protecting privacy. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, 10(5):557–570, 2002. Available from: `privacy.cs.cmu.edu/people/sweeney/kanonymity.pdf`.

[332] Peter P. Swire and Robert E. Litan. *None of Your Business: World Data Flows, Electronic Commerce, and the European Privacy Directive*. Brookings Institution Press, Washington, USA, online edition edition, 1998. As cited in [302]. Available from: `brookings.nap.edu/books/081578239X/html/`.

[333] Humphrey Taylor. The harris poll. Technical Report 17, HarrisInteractive, March 2003. Available from: www.harrisinteractive.com/harris_poll/index.asp?PID=365.

[334] TNS Emnid. TNS Emnid untersucht die Akzeptanz von Kundenkarten unter deutschen Verbrauchern. Consumer survey, April 2002. Available from: www.tns-emnid.com/2004/pdf/presse-presseinformationen/2002/2002_04_26_TNS_Emnid_Kundenkarten.pdf.

[335] United Nations. Universal declaration of human rights. Adopted and proclaimed by General Assembly resolution 217 A (III) of December 10, 1948. Available from: www.un.org/Overview/rights.html.

[336] Eric M. Uslaner. *Democracy and social captial*, chapter 5, pages 121–150. In Warren [344], 1999.

[337] Gilles W. van Blarkom, John J. Borking, and Eddy Olk, editors. *Handbook of Privacy and Privacy-Enhancing Technologies – The Case of Intelligent Software Agents*. College bescherming persoonsgegevens, The Hague, The Netherlands, 2003. Available from: www.andrewpatrick.ca/pisa/handbook/handbook.html.

[338] Hal R. Varian. The information economy: How much will two bits be worth in the digital marketplace? *Scientific American*, 273(3):200–2001, September 1995. Available from: www.sims.berkeley.edu/~hal/pages/sciam.html.

[339] Jennifer Vogel. When cards come collecting. *Seattle Weekly*, September 16–22, 1998. Available from: www.seattleweekly.com/features/9838/features-vogel.php.

[340] Harald Vogt. Efficient object identification with passive RFID tags. In Friedemann Mattern and Mahmoud Nagshineh, editors, *Proceedings of the First International Conference on Pervasive Computing (Pervasive 2002)*, volume 2414 of *LNCS*, pages 98–113. Springer-Verlag, 2002.

[341] Wolfgang Wahlster, Jörg Baus, Christian Kray, and Antonio Krüger. REAL: ein ressourcenadaptierendes mobiles Navigationssystem. *Informatik Forschung und Entwicklung*, 16(4):233–

241, 2001. Available from: link.springer.de/link/service/journals/00450/bibs/1016004/10160233.htm.

[342] Jim Waldo. The Jini Architecture for Network-centric Computing. *Communications of the ACM*, 42(7):76–82, July 1999.

[343] Mark Ward. A small slice of design. *BBC News*, April 2001. Available from: news.bbc.co.uk/1/hi/sci/tech/1264205.stm.

[344] Mark E. Warren, editor. *Democracy and Trust*. Cambridge University Press, Cambridge, UK, 1999.

[345] Samuel Warren and Louis Brandeis. The right to privacy. *Harvard Law Review*, 4(5):193 – 220, December 1890. Available from: www.lawrence.edu/fac/boardmaw/Privacy_brand_warr2.html.

[346] Marcel Wassmer. Konzeption und Implementation von Privacy Beacons. Master's thesis, ETH Zurich, Zurich, Switzerland, 2004.

[347] Duncan J. Watts and Steven H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, (393):440–442, 1998. Available from: www.nature.com/cgi-taf/DynaPage.taf?file=/nature/journal/v393/n6684/full/393440a0_fs.html.

[348] Thorsten Weber. RFID – Radio Frequency Identification. mEuro.info Website, 2005. Available from: www.myeuro.info/rfid.php.

[349] Stephen A. Weis. *Security and Privacy in Radio-Frequency Identification Devices*. Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, May 2003. Available from: theory.lcs.mit.edu/~sweis/.

[350] Stephen A. Weis, Sanjay E. Sarma, Ron L. Rivest, and Daniel W. Engels. Security and privacy aspects of low-cost radio frequency identification systems. In *Security in Pervasive Computing: First International Conference*, volume 2802 of *LNCS*, pages 201–212, Boppard, Germany, 2003. Springer-Verlag. Available from: www.springerlink.com/openurl.asp?genre=issue&issn=0302-9743&volume=2802.

[351] Mark Weiser. The computer for the 21st century. *Scientific American*, 265(3):66–75, September 1991. Reprinted in IEEE Perva-

sive Computing, 1(1), Jan.-Mar. 2002, pp. 19–25. Available from: www.ubiq.com/hypertext/weiser/SciAmDraft3.html.

[352] Mark Weiser. Some computer science issues in ubiquitous computing. *Communications of the ACM*, July 1993. Available from: nano.xerox.com/hypertext/weiser/UbiCACM.html.

[353] Henry R. West. Utilitarianism. In *Encyclopædia Britannica*. Deluxe Edition CD-ROM, 2004.

[354] Alan F. Westin. *Privacy and Freedom*. Atheneum, New York, USA, 1967.

[355] Cryptographic Hash Function (Nov. 30, 2004, 08:59 UTC). In *Wikipedia: The Free Encyclopedia*. Wikimedia Foundation, 2004. Available from: en.wikipedia.org/wiki/Message_digest.

[356] Enhanced 911 (Oct. 8, 2004, 02:53 UTC). In *Wikipedia: The Free Encyclopedia*. Wikimedia Foundation, 2004. Available from: en.wikipedia.org/wiki/E911.

[357] Felicific Calculus (Aug. 6, 2004, 23:45 UTC). In *Wikipedia: The Free Encyclopedia*. Wikimedia Foundation, 2004. Available from: en.wikipedia.org/wiki/Felicific_calculus.

[358] Jeremy Bentham (Aug. 21, 2004, 17:37 UTC). In *Wikipedia: The Free Encyclopedia*. Wikimedia Foundation, 2004. Available from: en.wikipedia.org/wiki/Jeremy_Bentham.

[359] Negative Liberty (Aug. 16, 2004, 19:04 UTC). In *Wikipedia: The Free Encyclopedia*. Wikimedia Foundation, 2004. Available from: en.wikipedia.org/wiki/Negative_liberty.

[360] Prisoner's Dilemma (Oct. 3, 2004, 13:52 UTC). In *Wikipedia: The Free Encyclopedia*. Wikimedia Foundation, 2004. Available from: en.wikipedia.org/wiki/Prisoners_dilemma.

[361] Public Key Cryptography (Dec. 24, 2004, 00:53 UTC). In *Wikipedia: The Free Encyclopedia*. Wikimedia Foundation, 2004. Available from: en.wikipedia.org/wiki/Public_key_cryptography.

[362] Transport Layer Security (Dec. 9, 2004, 21:47 UTC). In *Wikipedia: The Free Encyclopedia*. Wikimedia Foundation, 2004.

Available from: en.wikipedia.org/wiki/Secure_Sockets_Layer.

[363] United States Constitution (Feb. 10, 2004, 04:34 UTC). In *Wikipedia: The Free Encyclopedia*. Wikimedia Foundation, 2004. Available from: en.wikipedia.org/wiki/United_States_Constitution.

[364] Common Law (Feb. 9, 2005, 10:13 UTC). In *Wikipedia: The Free Encyclopedia*. Wikimedia Foundation, 2005. Available from: en.wikipedia.org/wiki/Common_law.

[365] Digital Divide (Feb. 20, 2005, 19:02 UTC). In *Wikipedia: The Free Encyclopedia*. Wikimedia Foundation, 2005. Available from: en.wikipedia.org/wiki/Digital_divide.

[366] Global Positioning System (Feb. 14, 2005, 13:49 UTC). In *Wikipedia: The Free Encyclopedia*. Wikimedia Foundation, 2005. Available from: en.wikipedia.org/wiki/Global_Positioning_System.

[367] Secretary of State for the Home Department (Feb. 25, 2005, 12:11 UTC). In *Wikipedia: The Free Encyclopedia*. Wikimedia Foundation, 2005. Available from: en.wikipedia.org/wiki/Home_secretary.

[368] Statute (Dec. 20, 2004, 00:31 UTC). In *Wikipedia: The Free Encyclopedia*. Wikimedia Foundation, 2005. Available from: en.wikipedia.org/wiki/Statute.

[369] Tort (Feb. 17, 2005, 03:13 UTC). In *Wikipedia: The Free Encyclopedia*. Wikimedia Foundation, 2005. Available from: en.wikipedia.org/wiki/Tort.

[370] Marianne Winslett, Ting Yu, Kent E. Seamons, Adam Hess, Jared Jacobson, Ryan Jarvis, Bryan Smith, and Lina Yu. Negotiating trust on the Web. *IEEE Internet Computing, Special Issue on the Technology of Trust*, 6(6):30–37, November 2002. Available from: www.computer.org/internet/ic2002/w6toc.htm.

[371] Love: Japanese style. *Wired News*, June 11, 1998. Available from: www.wired.com/news/culture/0,1284,12899,00.html.

[372] Sun on privacy: 'Get over it'. *Wired News*, January 26, 1999. Available from: www.wired.com/news/politics/0,1283,17538,00.htm.

[373] Hi, do you beam here often?, March 25, 2000. Available from: www.wired.com/news/technology/0,1282,35090,00.html.

[374] Can spam? Or new can of worms? *Wired News*, December 22, 2003. Available from: www.wired.com/news/politics/0,1283,61679,00.html.

[375] Due process vanishes in thin air. *Wired News*, April 8, 2003. Available from: www.wired.com/news/print/0,1294,58386,00.html.

[376] American passports to get chipped. *Wired News*, October 21, 2004. Available from: www.wired.com/news/privacy/0,1848,65412,00.html.

[377] Passport safety, privacy face off. *Wired News*, March 21, 2004. Available from: www.wired.com/news/privacy/0,1848,62876,00.html.

[378] Charles W. Wolfram. *Modern Legal Ethics*. West Publishing Company, Student edition, 1986.

[379] Philip Worchel. Trust and distrust. In William G. Austin and Stephen Worchel, editors, *The Social Psychology of Intergroup Relations*. Wadsworth, Belmont, CA, USA, 1979.

[380] Toshio Yamagishi and Midori Yamagishi. Trust and commitment in the United States and Japan. *Motivation and Emotion*, (18):129–166, 1994. As cited in [336].

[381] Junko Yoshida. Euro bank notes to embed RFID chips by 2005. *EETimes*, December 19, 2001. Available from: www.eetimes.com/story/OEG20011219S0016.

[382] Markus Zeidler. RFID: Der Schnüffelchip im Joghurtbecher. Monitor-Magazin, January 8, 2003. Available from: www.wdr.de/tv/monitor/beitrag.phtml?bid=554&sid=108.

[383] Alf Zugenmaier, Michael Kreuzer, and Günter Müller. The Freiburg privacy diamond: An attacker model for a mobile computing environment. In Klaus Irmscher and Klaus-Peter Fähnrich, editors, *KiVS Kurzbeiträge*, pages 131–141, Leipzig, Germany, February 2003. VDE Verlag.