IST Amigo Project
Deliverable D4.7

7 - Intelligent User Services

Privacy and Personal Security

IST-2004-004182

Public

Information Society
Technologies

Project Number	:	IST-004182
Project Title	:	Amigo
Deliverable Type	:	Report

Deliverable Number	:	D4.7
Title of Deliverable	:	7 - Amigo Intelligent User Services: Privacy and Personal Security
Nature of Deliverable	:	Public
Internal Document Number	:	amigo_7_d4.7_final
Contractual Delivery Date	:	30 November 2007
Actual Delivery Date	:	14 January 2008
Contributing WPs	:	WP4
Author(s)	:	Kamran Sheikh, Maarten Wegdam, Remco Poortinga (Telin) Maddy Janse, Iris Soute, Heleen Boland, Abdullah al Mahmud, LeeChin Yeo, Peter Vink (Philips)

Abstract

This document presents the way in which the Amigo project approached the complex problem of handling privacy and personal security in an ambient intelligent networked environment. Since protecting and maintaining privacy is a key user requirement, a user-driven approach was adopted. User studies were conducted to explore the different conditions and constraints that affect people's perceived privacy in a networked home environment. Context-aware applications for an extended home environment provided the setting and conditions for inducing privacy-sensitive situations. People's preferences for masking and hiding information that is being shared were investigated for different types of applications. The results showed that people use various mechanisms to preserve their privacy. They share their personal information only with a small group of close relatives and friends and only when there is a clear benefit for them. Maintaining control over the level of information that is being shared is crucial. Design guidelines were derived from these results to address end-users' privacy requirements. Next, an application was built for sharing different kinds of information between different users in different situations using Amigo services. This activity resulted in a model of the environment and of privacy related settings that are needed from the user perspective. The preferences for privacy control are dependent on the type of information that is being shared, the level of detail and with whom users are sharing this information. The settings that have to be made are context-aware and privacy-aware. Quality of Context (QoC) was investigated as a potential mechanism to limit sharing of context information. For this, a context aware privacy preferences collection and enforcement (CAPriCE) framework was developed. This framework enforces context aware privacy policies. These policies are based on templates provided by application developers and can optionally be refined and extended by users to match their privacy preferences.

Keyword list

Ambient intelligent systems, networked home environment, perceived privacy, context-aware applications, user studies, amigo intelligent user services, context management, user modeling, privacy, trust, context-awareness, presence sharing, extended home environment, connected home

Table of Contents

1 Introduction

This document reports how the Amigo project has addressed the complicated subject of perceived privacy and security. Previous deliverables have already reported on parts of the work. This final deliverable for the Perceived Privacy and Security subtask of the Amigo Intelligent User Services (WP4) comprises the user studies that were conducted to provide design guidelines, the development of an application prototype to refine the guidelines and to generate a model suitable for the Amigo environment, and a privacy policy framework to handle privacy policies.

The Amigo approach towards perceived privacy and personal security consists of 3 distinct phases. The objective of the first phase was to gain insight and an overall view of the problem. Here we took a user-centered approach and conducted user studies that resulted in design guidelines. In the second phase, an application prototype was developed to model the privacy for an Amigo environment and to validate and refine the design guidelines. In the third phase a privacy framework was set-up for handling user's privacy in context aware environments.

At the start of the Amigo project a Perceived Privacy and Security module was foreseen in the Intelligent User Services architecture. The field studies and the Amigo scenario evaluations that were conducted in the first months of the project, however, showed that to fulfill user requirements with regard to perceived privacy and its control, user's individual differences and preferences, and the situational dependencies was far too complex to handle without further exploration and a clear focus on an application. Hence, a subset of the Amigo scenario was used as the carrier, and implemented such that it could be used in a field study. The results from these studies provided design guidelines for addressing perceived privacy. Furthermore, these results also made clear that perceived privacy should be handled at the application and the middleware level. These studies are reported in Chapter 2.

Chapter 3 presents the user perceived privacy application that incorporates the design guidelines and integrates the Amigo Intelligent User Services, i.e., the Context Management Service (CMS) and the User Modeling and Profiling Service (UMPS). The software that is needed, in addition to the CMS and UMPS for this application is available on gforge.

Chapter 4 presents the privacy policy framework, CAPriCE, a context aware privacy preferences collection and enforcement framework. This framework reduces the burden for the end-users and supports application and context management framework developers by setting default privacy configurations. By using policy templates, application specific defaults can be provided. This privacy policy framework is implemented as a rule-based system. The software is available on gforge.

The Extended Home Environment scenario from Amigo WP7 was used as a basis for the field studies, the application implementation and the development of the privacy policy framework. This Extended Home Environment is particularly vulnerable. To ensure people's privacy in such an environment implies accounting for the implications induced by acquiring, collecting and inferring personal information of users. Tracking and collecting significant portions of users' everyday activities and interactions are required to compose user profiles and to model the context in which these user behavior's and interactions occur. Disclosure of such private information to other parties, whether it be friends or family, service providers or commercial traders, in return for benefits like receiving a desired personalized and context-aware service or conducting specific actions, induces a delicate balance that needs to be maintained [1] and protected. Furthermore, home environments are dynamic environments in which conditions continuously change and consequently the users' requirements with regard to their privacy change as well.

In sum, perceived privacy is a complicated subject that has to be dealt with at various points in the software architecture. Handling privacy differs per user, application and situation (context).

Because of this complexity, the approach of the Amigo privacy task team has been to first conduct end-user studies to derive guidelines on how to build privacy-aware applications from the end-user's point of view. At the same time, research has been done into the so-called *Quality of Context* (QoC) area as potential mechanism to limit sharing of context information. These findings were mapped onto the Amigo extended home area and led to the development of a test application that uses the retrieved data in a privacy sensitive way, based on local context.

The final result of the PPS task within Amigo consists broadly of 3 different components:

1. A set of (derived) perceived user privacy guidelines
2. An example application for validation of these guidelines.
3. A Context Aware Privacy preferences Collection and Enforcement framework (CAPriCE).

The general attributes of the two software components are described in the two following sections below, while the detailed description of these components as well as the research into user perceived privacy are handled in detail in the following chapters.

1.1 Example user perceived privacy application

Provider

Philips

Introduction

In order to perform, the second phase of our studies, a simple mechanism was added to the existing 'shared activities' demonstration in the extended home domain. This demonstrator has been written in Javascript and therefore this was also chosen for the privacy related extensions. In summary, the following extensions were made.

- Extension to the CMS services to support Precision in the ontology, introduction of privacy levels.
- Scripts to access CMS in order to make the application context aware.
- Scripts to access UMPS for retrieving Privacy related settings
- A simple GUI, embedded in the demonstrator to edit some settings
- A colored light, available as an amigo service, used to provide user feedback on privacy related information sharing
- Scripts to access this light

Development status

Since the shared activities demonstrator is not a public deliverable, it was also not possible to provide the embedded GUI. However, the scripts have been made available and writing a different GUI is quite straightforward.

Intended audience

Developers

Licence

Philips Licence, BSD style

Language

Javascript

Environment (set-up) info

Scripts can be embedded in javascript or html.

Platform Hardware

Tested for firefox browser

1.2 CAPriCE framework

Provider

Telin

Introduction

The Context Aware Privacy preferences Collection and Enforcement (CAPriCE) framework is a framework for enforcing context aware privacy policies. The policies are based on templates provided by application developers; these policies can optionally be refined and extended by individual users to match their privacy preferences. The main part consists of an adapted XACML engine, with context management framework specific Policy Enforcement Points.

Development status

The final CAPriCE version is available from gforge.

Intended audience

Developers of context management frameworks and context aware applications

License

Sun specific license, see http://sunxacml.sourceforge.net/license.txt

Language

Java

Environment (set-up) info needed if you want to run this sw (service)

PC with JVM

Platform

JVM

1.3 Scenario used for the privacy and personal security activities

The following scenario excerpts are taken from the overall Amigo scenario. They are the basis for the work on privacy and personal security. The excerpts are given to facilitate reading and referencing in this document (See Amigo Deliverable D1.2. Volume-I Summary for full deliverable).

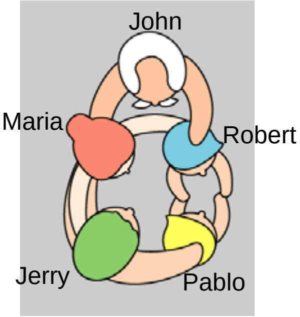
Setting

Maria moved recently to Eindhoven with her husband Jerry and their two sons Roberto and Pablo.

Before moving to *Eindhoven*, Maria and her family were living with her father, John, in *Brussels.* John is living alone.

Maria and her father have installed Amigo systems in their houses to maintain close contact and still feel like being part of each other's daily family life.

Their Amigo systems also help them with the daily housekeeping chores, their social agendas and taking care of their news and entertainment needs.



Extended Home Environment

Roberto and his grandfather John have continued their habit of playing games together, watching a bit of TV and having their man-to-man chat. Amigo takes care of setting up the right ambiance. John's Amigo system is a modest version, with which he can be a participant in Roberto's games, just like Roberto's peers. Amigo selects the games that both John and Roberto like. They can look at each other and see what game moves are being made. Amigo can also set-up a video-conference for them in which they can watch TV together, show the newest acquisitions of their collections, or just tell their stories. With Pablo, the little one, John plays 'hide-and-seek' via this video communication.

2 User Derived Design Guidelines

Applications in extended networked home environments are intended to facilitate the communication between users of different households and to provide them with a feeling of a shared ambiance. Connecting people in this way influences their social relationships. Home, as we know it, is a place where people can retreat from society and its social rules. Extended home applications induce intrusions in this familiar and trusted environment. Since ambient intelligent systems are by definition unobtrusive and embedded in the user's environment, users might easily forget their existence and unwillingly have their privacy violated. "Perceived privacy" or how end-users perceive that the system affects their privacy is one of the key aspects for the acceptance of ambient intelligent systems by users. It is also one of the most complex problems to handle. It is about 'how, when, and to what extent' data about people are revealed to other people within a dynamic social context[1].

An empirical approach was taken to address this problem in which exploratory, field and concept studies were conducted to acquire user input to derive design guidelines for application development and for specific application implementations. The context in which these studies were conducted was derived from the application scenarios for the networked extended home environment. In the first study privacy handling by users in everyday communications was explored. This study was followed by a field experiment in which people's actual behavior in an extended home situation was observed. The third study addressed perceived privacy in specific application dependent situations. These studies are presented in this chapter.

2.1 Privacy handling in everyday communications

An exploratory study was conducted to obtain implicit information about people's attitudes towards privacy sensitive situations that might be induced by having an operational networked extended home environment. The most commonly used media types, such as, mobile phone, home phone, MSN, e-mail and regular mail were selected as target and classified according to the different contact categories with regard to their content: practical, social, emotional and special occasion. This structure was used to address the following questions: what are the privacy needs for the different types of communication media and how do people currently handle their privacy needs?

2.1.1 Methodology

To acquire information about how people handle their privacy in everyday communications, an ethnographic methodology was used in which people were asked to keep a diary for one week to record all their home-based communications. After that period a semi-structured interview was conducted. The diary served to log all types of one week's communications and to facilitate the interview by having explicit cues available. It excluded face-to-face meetings and communications outside the home. The interview focused on what types of information people regard as highly sensitive and how they make sure that their moments of communication are not disrupted. A storyboard for guided exploration was used to structure the interview. The storyboard presented different potentially privacy violating situations, for example, presence notification, automatic identification and automatic intervention. To accomplish a comprehensive coverage of everyday communication, participants were selected from a wide

[1] Amigo Deliverable D4-5 presented related work and background. The references are included in this document in the References chapter.

range of age and social situation. The participants (n=6) were: (a) an 88 year-old grandmother with a large family of children and grandchildren and living with her husband; (b) two middle-aged men (39 and 41) with a young family; (c) a woman, aged 25, in a single household; (d) one man, aged 27, living with a girlfriend; (e) a teenage girl (age 15) living with her parents and a sister. This sample of participants covering a wide range of social conditions was used to explore the handling of privacy in daily communication situations in a qualitative way.

2.1.2 Results and conclusions

Privacy sensitive communications are usually personal and/or emotional according to most participants. Examples of disruptive situations for them are: someone at the door, someone on the other phone line, bad telephone connections and noisy children in the house. Escaping to the attic or keeping children busy with a movie was used as a strategy to maintain privacy. The participants' opinions of the location and presence awareness system that was presented in the storyboard scenario varied a lot. The family men saw no use for it; they would not share their presence and location with others, only maybe for staying in touch with their family when traveling. The grandmother doubted the usability and usefulness of the new technology. The single woman would like to know the availability for communication of her family and friends, but she would not share her availability with them. That is, an asymmetric attitude towards the information. The teenage girl didn't trust the privacy protection of the system. Automatic identification wasn't considered as a privacy risk. People assumed that if they would use it, they would also know the privacy risks. Automatic presence notification was an ambiguous concept for the participants. They would prefer to set their own presence and availability for communication, but they also acknowledge that this would require too much effort. Also, they didn't trust others to set their presence and availability. The young adults (in their twenties) were frequent users of MSN and Skype programs, but they rarely used the availability information. As message senders they ignored the status information as it is not always accurate because it is automatically set to 'away' when the user is not active on the computer. As message recipients they used the status information to ignore incoming messages in a socially acceptable way. Automatic intervention to protect user's privacy was considered neither useful nor desirable. According to the participants, it is rude, asocial and inconsiderate to shut off communication automatically without warning all the users involved.

In sum, the most important ways in which people handle their privacy is to isolate themselves from other family members and outside interruptions. To achieve such isolation, they retreat to private rooms or have agreements for not being disturbed. They also use plausible excuses for not communicating, for example, 'failure of technology' to mask their real reasons like 'not being in the mood to communicate'. Their strategies are rather ego-centric as they are more appreciative of being able to see someone else's presence or availability than showing their own presence. They only tend to see the implications of their privacy settings, but not what the implications of these settings are for other people.

2.2 Sharing presence information: In the field

Isolation appears to be one of the most important strategies for people to protect their privacy in familiar everyday communication situations. To understand how this behavior in actual privacy sensitive situations is affected, a field study was conducted in which presence information was shared between two connected homes. A functional prototype was placed in the homes that could be used for 2 weeks and for which the experiences of the users could be investigated. Perceived privacy was measured with questionnaires that addressed five composite concepts: perceived social presence (4), perceived control (5), perceived effort, perceived connectedness and social presence (6). Automatic and manual presence detection conditions were used.

2.2.1 Methodology

Four pairs of friends/relatives participated in the study; one person per household. Two of the participant pairs had a parent-child relationship (children: mid-twenties; parents: mid-fifty), one pair were sisters (mid-twenties) and one pair were very close friends (mid-twenties). The two households were connected by two HomeLamp systems that showed the presence of persons in their homes. The systems supported a basic form of location-tracking to detect whether a person was at home or not. The HomeLamp-system consisted of a small-form-factor computer, an amBX[2]-lamp and a sensing device for detecting wireless tags (7). The tags were attached to a key ring. They had a button for toggling between 'present' and 'not present' status. Users could set the presence status in the manual condition and override it in the automatic status. The range of detection was 300m. When people entered or left their home, their presence was detected by the system and shown by the lamp. The amBX-lamp generated different color patterns. Each participant had a personal color to indicate presence status. When a participant was at home, then this was also shown in the other home by light and color indication (Figure 2-1). The systems were permanently connected to the Internet and the presence information was shared by using the Jabber protocol. The information that is shown on one system is identical to the information that is shown on the other, connected system.

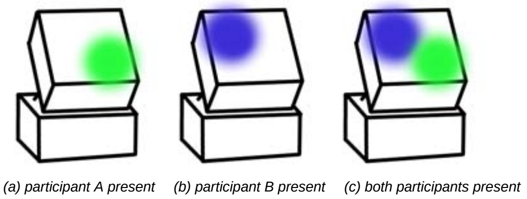

(a) participant A present (b) participant B present (c) both participants present

Figure 2-1: Presence status of participant A (light grey) and participant B (dark grey) as indicated by the lamp

2.2.2 Field study results

The results of the questionnaires were summarized over all participants and analyzed separately for social presence, connectedness, social privacy and control and effort. In addition, the reliability of the rating scales was measured and if it was sufficiently high non-parametric tests were used to test the difference between the manual and automatic conditions. The 'feeling of social presence', was rated significantly higher in the automatic condition (mean rating 5.1) than in the manual condition (mean rating 4.5) on a scale of 1 (least) to 7 (most). The concepts 'connectedness' and 'social privacy' were measured by means of 5 separate items: Expectations, Invasion of privacy, Obligations, Sharing experiences, Staying in touch, Thinking about each other. The ratings for these separate items showed large variability, meaning that the participants did not agree on them.

The items for invasion of privacy are rated higher than the items for feelings of expectation and obligation. Thinking about each other and staying in touch are rated higher than sharing experiences. These differences were, however, not significant. The items concerning effort had slightly higher ratings with less variability across participants than the items concerning control. There were no clear differences between the automatic condition and the manual condition except for the amount of attention that the system required.

[2] © Philips

The results of the interviews were analyzed and clustered based on consensus between two independent analyses of the audio data. The most salient groupings are reported here. They concern aspects of ‘perceived effort and control’ and ‘perceived sharing and connectedness’.

2.2.3 Perceived effort and control

People preferred the automatic condition over the manual condition. According to them, it would take too much effort; they had to think about it and conduct an intentional action to show being at home. Some participants wanted to be in control of the HomeLamp, irrespective of whether they used that control or not. Agreements were made between participants on how to use the HomeLamp. They preferred to show their availability rather than their presence, if it wouldn’t take too much effort to do so. The presence information was only shared with a very small group of close relatives and friends and didn’t go beyond sharing more information than their availability or presence in the house. Detailed location information as well as detailed activity information was considered to be too privacy sensitive. The information in the manual condition was considered less reliable than the information in the automatic condition because participants occasionally forgot to turn the HomeLamp on (in the manual condition) and they also expected the other party to forget it as well.

2.2.4 Perceived sharing and connectedness

Most participants preferred a social solution over using the system for sharing their availability. They preferred to simply tell the other person that they were not available instead of using the system to show this. The HomeLamp increased a feeling of connectedness, which was in most cases a positive feeling. However, for the parent-child relationships it felt sometimes as an overload of information. Children felt being monitored and parents became anxious when their child, for example, was not present or did not answer a call when expected. Friends or sibling pairs did not associate negative feelings with such unexpected situations. Multi-person situations between household members did not pose specific privacy issues for the participants. The children in the parent-child relation used deception. They felt uncomfortable about hiding information, but they also felt forced to use such deception.

In addition to the responses of the participants to the interview questions, specific observations were made. First, everybody has a different kind of Internet connection and households with more than one person usually have one person who is the administrator. Second, people didn’t feel monitored in the automatic condition. The only comparative comments for the manual and the automatic conditions concerned the difference in effort and the difference in reliability. Third, the behavior of the participants in the manual condition showed that they became more casual with regard to turning the lamp on or off when they changed their presence situation.

2.2.5 Conclusions

For most participants the HomeLamp definitely increased their feeling of connectedness. But, this benefit also depended on how they normally keep in touch. As for the conditions, the manual condition took too much effort. The feeling of social presence is higher for the automatic than for the manual condition. Arguably, this might be connected with the fact that the perceived reliability of the manual presence indication was low because participants sometimes forgot to use the system.

Sharing information about being home is about as detailed as the participants liked it. More detailed information was generally considered to be too privacy sensitive. However, this remains a matter of subjective preferences and depends on how well the other can interpret the information. Participants only wanted to share their location information with a small group of people. Sharing information might not only have a negative effect on the sharer, but also on the receiver. Too much information might lead to anxiety, especially if the receiver has some sort of caring function.

In short, this field study showed that people will share their information with only a small group of close relatives and friends, the sharing of the location information should have a clear benefit, users need a feeling of being in control and the desired level of detail of the location information is subjective. Furthermore, the large variability in the behavior of people in privacy sensitive situations induced by, for example, presence sharing applications has implications for the design and use of awareness systems. Individual differences, varying social relations and application specifics have to be considered.

2.3 Maintaining privacy in application specific situations: Using different types of noise

To investigate what people do to maintain their privacy in application-specific privacy sensitive situations, a conceptual design study was conducted. The focus of this study was on how people want to hide information that is being shared to maintain their privacy. One of the most important conclusions from the HomeLamp field study was that people want control over the level of detail in which their information is shared. Price et al. (8) propose a model for user control of privacy that incorporates 'noise' by introducing ambiguities in the information. This model deals with location and identity information and is divided into 5 types of 'noise':

- Anonymity: hide identity.
- Hashing: disguise identity.
- Cloaking: be invisible.
- Blurring: decrease accuracy of data
- Lying: give intentionally false information

The conceptual design study investigated how well these noise forms fit extended home environment applications and which noise forms are desired by the users. The following application concepts were investigated: 1) sharing photos, 2) sharing location information, and 3) sharing health information. People's perceived privacy is influenced by how they appreciate the usefulness of presence sharing information. These applications were selected because it is assumed that their perceived usefulness differs.

2.3.1 Methodology

The information that could possibly be privacy sensitive was identified for each application concept and noise forms from (8) were adapted to each of them. Participants were shown sketches of these concepts. The participants (N=18, age 25-52 yrs.) were asked to evaluate the three application concepts, indicate in what form they would want to share their data, with whom they would like to share and in which context they would use it. Three tasks were carried out for each application and participants had to indicate and explain which noise forms they would: (a) optionally want in the proposed application, (b) have as their default setting, and (c) rank highest regarding perceived importance. A thinking-aloud methodology was used (9). Cards were used to present each information type. An example of the task presentation and its setting is shown in (Figure 2-2).

2.3.2 Results

The photo sharing application was preferred the most useful, followed by sharing health information. Least preferred was sharing location information. The preferences for noise forms differed per application. Correlation between rankings was calculated using the Kendall coefficient of concordance.

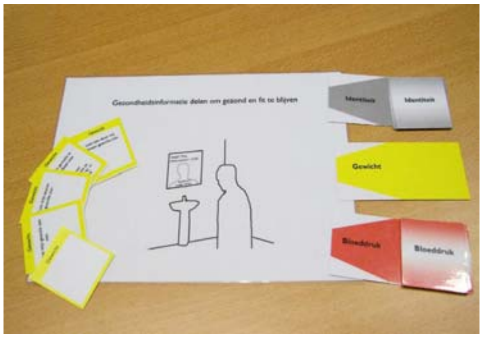


Figure 2-3: Example of task assignment for the conceptual design study. This application can unobtrusively monitor your health, for example your blood pressure and weight, but also other health factors can be measured. Each morning the information is collected and sent for example to your physician, or to your sport coach, etc. How would you handle these situations and which noise forms (exact data, less detail, free data choice, no show) would you select in each situation?

The ranking results are shown in Table 2-1. The rankings for the photo sharing application showed that no background video and voice only is preferred over blurring the background video and distorting the audio. Regarding identity, the use of nicknames (hashing) and incomplete identity (blurring) was preferred over lying (fake identity). Incomplete identity (blurring) was liked the most and showing no identity (be anonymous) the least for the location sharing application. Showing their location approximately (blurring) was preferred over giving a fake location (lying). For the location information sharing application settings there was less agreement amongst the participants then for the photo sharing application settings. Agreement among participants was highest for the health information sharing application. Sharing accurate weight and blood pressure (no noise) was preferred to giving a fake weight and blood pressure (lying). With regard to the health information sharing application, people preferred to use their accurate and complete identity and medical data (no noise). Lying about these data was not liked.

Table 2-1: Ranking results for Photo sharing, Location sharing and Health sharing applications as participants' choices (n=18).

Noise form	Card sorting choice	Default choice*
	Photo Sharing Application	
Identity		
• Accurate and complete	9	5
• Incomplete	15	9
• Nick name	13	4
• Fake identity	8	0
• No identity	6	0
Video		
• Full video (person + background)	17	4
• Blurred video	6	0
• No background (person only)	13	7
• No video	12	4
Audio		

• Full audio	14	5
• Voice only	17	11
• Fade voice	2	0
• Distort voice	5	0
• No audio	10	1
	Location Sharing Application	
Identity		
• Accurate and complete	9	5
• Incomplete	15	9
• Nick name	13	4
• Fake identity	8	0
• No identity	6	0
Location		
• Accurate location	12	3
• Approximate detail	17	11
• Fake location	9	0
• No location	13	3
	Health Sharing Application	
Identity		
• Accurate and complete	15	9
• Incomplete	10	3
• Nick name	6	3
• Fake identity	4	1
• No identity	8	2
Blood pressure		
• Accurate blood pressure	18	10
• Approximate blood pressure	11	4
• Fake blood pressure	1	0
• No blood pressure	8	3
Weight		
• Accurate weight	18	10
• Approximate weight	8	4
• Fake weight	1	0
• No weight	10	3

* Not all participants could make a choice

2.3.3 Conclusions

The highest level of agreement was found for the use of full video and audio as noise in the photo sharing application and for accurate weight and blood pressure data in the health application. That is, people don’t want to protect their privacy by blurring video and audio backgrounds if they share photos and they don’t want to lie about their blood pressure and weight if it is for their health’s benefit. The lowest level of agreement was found for using identity information in the photo sharing and health information sharing applications. In other words, there is no common preference for a noise type to mask identity information.

Although there was no decisive result, it appears that people in general prefer to give less detailed information over giving accurate information. This also confirms the findings from the HomeLamp field study. Participants provided a multitude of contexts for which they would use different noise forms to maintain social norms and values. Preferences for noise forms differed per application. These contexts are quite complex, as they dynamically change depending on with whom the information is shared and on the perceived benefits for the user.

In general people prefer to share information at the lowest level of detail that is appropriate for the application, but they also desire to add noise (for example lying) to comply with social acceptability. This implies that it should be easy for users of an Amigo like system to switch between noise forms depending on the dynamic nature of the context.

2.4 System design implications

The user research into perceived privacy started with an exploratory study, followed by a targeted study that involved limited implication effects and continued by studying different application specific contexts. Within this approach, the scope of the studies was limited to the extended home environment. In the exploratory study, people were asked about their daily communications and related privacy issues. In the field study, a system for presence detection in the home and sharing that information across homes was developed and evaluated. The conceptual design study was conducted to find out how people would like to be able to mask or hide information that is being shared between different parties for three different types of applications.

Main conclusions from these studies are:

- people use many diverse mechanisms to preserve their social privacy,
- people will share their personal information only with a small group of close relatives and friends,
- information sharing should have a clear benefit for users,
- users should have the possibility to control the level of detail of the information that is being shared, and
- users need a feeling of being in control, for example, automatic location detection is appreciated by users, but they also need to be able to influence the automatic detection mechanism.

Although the qualitative and quantitative results and observations from the user studies provided a wealth of information on the perceived privacy of users, they didn’t provide information on how data should be secured, stored, or encrypted to support and advice application development. Initially, it was proposed to handle privacy at the middleware service level of the Amigo architecture by means of a rule-based filter that incorporated the user’s preferences and that would use the preferences to either pass on the data or not. The field and concept studies showed, however, that such a mechanism for privacy filtering on the Amigo services level is not sufficient. Although there is definitely a need for a component that handles and stores the user’s privacy preferences, it is not sufficient for protecting the user’s perceived privacy, because it does not offer direct user control. Privacy should also be handled at application level. In particular, the type of information that is shared, the level of detail in which the information is shared, and with whom the information is shared (for example, with groups or individuals), are the most important concepts to take into account.

In addition to the implications for the system architecture, design guidelines were derived from the results of the qualitative and quantitative studies to support the development of extended home environment applications.

2.5 Design guidelines

First of all, the most important design rule to take into account is:

'Maximize user benefit, minimize user effort and provide reasonable control for the user'.
This implies maintaining a tricky balance between user benefit, effort and control. In addition to this rule, the following design guidelines need to be accounted for:

1. Provide proper security to prevent misuse of information and inform users of security measures.
2. Provide control to users over privacy settings at several levels of increasing complexity.
3. Present the user with a choice of level of detail in which the information should be shared
4. Provide clear feedback over shared information
5. Ask for user consent before sharing information
6. Avoid using automatic intervention to maintain user privacy.

These design guidelines are specifically for applications in the extended home environment and focus on the sharing of data in a social context. They are worked out in the following tables by means of a detailed description, a problem statement, an example, and a validation from the user studies.

Design guideline #1	**Provide proper security to prevent misuse of information and inform users of security measures**
Detailed description:	Provide technical solutions such as data encryption for protecting user information. Inform the end users of the security risks and solutions.
General problem:	If information is not shared in a secure way, the risk of misuse of that information is high. Misuse of information is very damaging to the acceptance of a service or a system by the end user because it violates the trust that users necessarily put in the system. Also, many users will not be able to assess the true security risks and either under-or overestimate them, so they should be informed of the true security risks.
Example:	If Maria and Jerry (the main actors in the Amigo scenario) and their children share location information and this information is not properly secured, there is a risk of burglars using it to their advantage.
Validation:	The HomeLamp conclusions show that all participants had a fear for intentional misuse of information by burglars. This issue also came up in the exploratory study and the conceptual design study.

Design guideline #2	**Provide control on several levels**
Detailed description:	Provide control over privacy settings on several levels, with increasing complexity. • a 'suspend-all' button • on the application level in two forms: (a) direct control, and (b) application-specific configuration • central level amigo system for configuration across applications
General problem:	If users do not feel in control over how information about them is shared, they will desire a higher level of privacy and consequently they will perceive sharing of information more easy as a privacy breach. The effort for controlling privacy in everyday life situations must be low, and even lower for emergency situations. Therefore, a suspend-all button must be provided, to enable the user to easily stop sharing of information. Since this is too rigorous for most common situations, each application needs to provide this control for the user for switching between different levels of detail (see DG 3). Furthermore, users need to be able to set defaults per application (configuration). For most users these two levels of control will be sufficient. For the group of more privacy conscious users the most complex control over a user's privacy can be dealt with on a central Amigo level.

Design guideline #2	**Provide control on several levels**
Example:	On the application level, Maria has configured the video and audio chat application such that when she is exercising with John (her father who is in another home), she shares full audio and video. However, when John is interrupted because somebody is at the door, Maria turns off the camera to protect her privacy in case the visitor decides to come into the room. She knows that if for some reason she does not want the visitor to know she is exercising, she can turn off the application all together by pushing one button.
Validation:	Participants in the user studies expressed the need for control to maintain their privacy. When they felt in control of their privacy (i.e., in controlling what part of their private data they could share with other persons), they were less inclined to be anxious for privacy breaches.

Design guideline #3	**Present the user with a choice of level of detail in which the information should be shared**
Detailed description:	Each type of information can be shared in several levels of detail and it should be possible for the user to adapt the level of detail to the context in which the information is shared.
General problem:	Although sharing information in the most exact way can sometimes be useful, users often feel a breach in privacy when they are forced to share their exact information all the time.
Example:	John and Maria share information about their physical condition, such as blood pressure and weight, while exercising. However, they only share whether the information is above or below the threshold. This way they can motivate and warn each other, but they don't feel monitored by each other.
Validation:	The participants in the field study felt comfortable with the system when it registered and shared the information about whether they were at home or not. Sharing detailed information, e.g. sharing information about in which room they were located, however, users felt that they were monitored and were not inclined to use the system: "Suppose that the system would show [my friend] that I was in my bedroom for an hour during the day, what would she think?! No, that's too much information."

Design guideline #4	**Provide clear feedback over shared information**
Detailed description:	Give an overview of all information that is being shared and in what form. Present this overview only at the user's request in an easy and quick way. Include the *what*, with *whom*, through what *medium*, and in which *level* of detail. Show the information exactly in the same way as the person with whom the information is shared receives it. The feedback must be real time and accurate. Indicate clearly if information might be less accurate. The feedback must be according to good usability practice.
General problem:	If users do not get proper feedback about how their information is shared, they will not feel in control of that information and they will lose their trust in the system. Consequently, users will desire a higher level of privacy and they will be very reluctant about sharing information.
Example:	When Maria is using video chat with John, a small portion of the screen is reserved for showing Maria what image of her is transmitted. If Maria turns the angle of the camera or uses the zoom functions she sees exactly what that does to her image and what John is seeing.
Validation:	Giving proper feedback through the user interface is always a must. For example, the Home-Lamps in the field study gave feedback about the users' status. This was appreciated.

Design guideline #5	**Ask for user consent before sharing information**
Detailed description:	Request the user’s consent for every new instance of information sharing (adding a receiver to a list of receivers or adding new information to a list of information) Request the user’s consent for making default settings for shared data. Avoid overwhelming the user with requests for consent
General problem:	Having no user consent increases the risk that users stop sharing information in order to maintain their privacy. It might happen that the information sharing (a) has already caused a privacy breach and (b) forces the user to become aware of his need to hide the information. Both situations are undesirable; a privacy breach needs to be prevented in all cases and being aware of hiding information can cause users to feel uncomfortable or guilty.
Example:	Roberto often calls on John (his grandfather) to play a game together. When they play games, they allow Amigo to set the ambiance. After the third time, Amigo suggests to John to set the ambience by default when Roberto calls and John can accept or decline the suggestion.
Validation:	The participants in the user studies felt guilty when lying about their presence status. They frequently referred to the importance of being in control over who receives what information.

Design guideline #6	**Avoid using automatic intervention to maintain user privacy**
Detailed description:	Avoid stopping sharing information automatically without consent of the user.
General problem:	Automatic intervention can cause offence, anxiety or snoopiness by undermining socially accepted behavior. Social solutions for maintaining privacy, such as excuses or not telling the whole truth are used by people to prevent such situations. These social solutions deal with delicate socially acceptable behaviors, which are subjective and situation-dependent. Automatic intervention cannot deal with these delicate socially accepted behaviors.
Example:	Maria and John are exercising and John receives a visitor. If Amigo were to interrupt the video chat automatically, Maria might not know what happened and feel anxious about John’s condition.
Validation:	According to the participants in the user studies it is not acceptable to automatically shut down video and audio channels when someone unexpectedly enters the room. This was considered rude and would cause anxiety. This also confirms one of the findings from the user requirements analysis of Amigo, i.e., ”don’t replace the individuals or make decisions for them regarding direct or indirect social relations”

In sum, the guidelines address the following aspects: levels of security, means for end-user control, levels of detail of the shared information, types of feedback to the end user, appropriateness for automatic sharing of information, and possibilities for automatic intervention for maintaining privacy. These guidelines are formulated in such a way that they can be used by system developers to create services and applications that are privacy-safe from an end-user’s point of view.

To show how these design guidelines could be used an extended home application was developed. This application is described in the next chapter. Furthermore, since these guidelines can also to a certain level be transformed in rules and default settings, a rule-based framework was designed to facilitate and support the development of applications. This framework is presented in Chapter 4.

3 Design Rules - Application Implementation

3.1 Introduction

The selected end-user application for this implementation fits a subset of the Amigo extended home environment scenario. In this application users have the possibility to share content and context information between two different physical locations. Photos are used as content in this application; for context information, users can share their locations and/or activity. .With regard to representing context information, different solutions were used, i.e., a light system in the living room and on-screen display options. These solutions were chosen because they could provide an intuitive, subtle and unobtrusive way to present context information.

The goal is to build an application that shares different kinds of information between different users in different situations using (some of the) Amigo services as defined in other work packages. This activity will then result in a model of the environment and of privacy related settings that are needed (and understood) from the user perspective. The preferences for privacy control, i.e., the settings that have to be made, are dependent on the type of information that is being shared, the level of detail and with whom users are sharing this information. This means that they are context-aware and privacy-aware.

3.2 User-centered design approach

The user-centered design methodology was used for the development of the application in which the user-derived guidelines and the user requirements from the Amigo project were taken as starting point. A scenario was derived from the Amigo Extended Home Environment scenario to represent the situation in which the intended technology and concept will be used. This phase was followed by the conceptual design phase and then the prototype implementation. The conceptual design included the user interface, the sharing application, means for representing context information and the privacy model, i.e., how to control data for protecting and preserving end user's privacy. A functional prototype was developed and integrated with the Amigo middleware services. The development of the prototype was done in an iterative manner by using intermediate expert evaluation results.

3.2.1 Scenario

The following scenario was used.

> The actors: *Maria and Jerry, a married couple and their 6 year old son Roberto.*
>
> The context: *Jerry travels a lot for his work. Some-times he is away for a month. Jerry and Maria use the Amigo extended home environment to keep in touch with each other and to maintain their homely ambiance*.
>
> The system: *the system can be used from any display device that is connected to the Internet. By using the system, Maria and Jerry can be together, while they are at different locations. They can exchange information, share photos, play games, chat, share activities, feel each other's presence and share their social context. They can do all these things at the same time, as if they are close together. Moreover, they can intuitively share their social context by using different light colors.*
>
> The script: *Maria and Jerry like to take photos of the special cocktails that they prepare. They like to talk about recipes and cocktails. But, they never talk about such things when Roberto is around.*
>
> *Maria has three different sets of new photos that she wants to upload to the system. The first set of photos is 10 years old. These are photos of her first marriage taken at the wedding day. She does not want anyone to see these photos. The second set of photos is recent. These are photos of their family visit to the zoo. She wants to look at these photos with Jerry and Roberto when they are together. The third set of photos shows the cocktails that she prepared a few days ago for their friends. She wants to look at these photos together with Jerry, but not with Roberto. Maria can tell the system these preferences. The system shows then the right photos with the right person at the right time*.

This month, Jerry is on a business trip in the US. Today is Sunday and Maria and Jerry have the day off. Maria likes to be with Jerry and share the photos that she took a few days ago when some friends visited her. Roberto is with Maria (the second set of photos (zoo) is shown on the display device). When it is Roberto's bedtime Maria brings him to his bedroom. When she returns to the living room, the photo set changes to the cocktail set on Maria's and Jerry's displays. In addition, the color of the light in Jerry's room changes at the same time to indicate that Maria is alone in the room.

Technical description of the scenario

Location A and location B each have an Amigo sharing application installed. This application could be installed in any display device like a television, a mobile phone, a computer or a PDA.

Maria and Roberto are in location A. Maria wants to share photos with Jerry, who is in location B. Maria logs in to her sharing application. The application shows her availability (for example, is online, is busy, is away or is offline) on both the display devices. At the same time, the light color in Jerry's location changes. Maria's availability and the availability of Jerry are checked by the intelligent system that is installed at both Amigo locations. It shows Maria that Jerry is present and available and it initiates a sharing request. The system in location B informs Jerry about the request from Maria. Jerry accepts the request. The sharing starts by showing a slide show of Maria's photos, which are set for sharing when Roberto is present. At the same time, Maria and Jerry can chat via text messaging or an audio/voice-chat service provided by the application.

When Roberto, the son of Maria and Jerry leaves the room in location A, while the sharing session is still going on, the Amigo system recognizes this change in context and forwards the information to the application. The application checks Maria's privacy policy from her policy database to know how to react when her child leaves the living room. In this case, Maria wants a different set of photos to be shown. The application reacts according to her privacy setting and shows the other set of photos in both locations. At the same time, Amigo in Jerry's location notifies Jerry's application about the changes by changing the light color in his location. The information represented by the light color is understood by Jerry based on a social agreement between Jerry and Maria.

There are two scenes in the scenario; one represents a static situation and the other a dynamic situation. In the static situation, Roberto is together with Maria when she initiates the sharing request. When Roberto leaves the room, it reflects the dynamic situation in which the context of Maria has changed. In both situations, Maria's privacy preferences should be handled by the application and the decision from the application should be context dependent.

3.3 Conceptual design

In this section the various aspects that play a role when privacy sensitive data are shared between users and homes are modeled. This model comprises (a) the basic concept of a privacy sensitive application and its context aware aspects, (b) the components that play a role in the Amigo network that is used for our application, (c) the data is to be shared and the possibility to filter or adapt it, and (d) the data that could influences a user's decision and therefore influences the structure of the rules/policies. Several design constraints and requirements needed to be accounted for.

1. To build a privacy aware and context aware application in an extended home environment.
2. The context information is retrieved from the environment.
3. Different control points are used to allow users to control which piece of information they are sharing with their contacts and how that information is shared.
4. The application should also use the context information for allowing the context in a user defined setting (e.g. sharing depends on context). In this way context information is not just regarded as data that is to be shared but also as data that can be used to make the privacy settings more powerful.
5. The application should allow users to share information and activities with their contacts at other locations. This information includes content, for example the user's photos, context information, for example, the user's exact or approximate location, and the user's social presence.

6. Feedback regarding the content and context changes should be represented in different ways, for example, by different icons and/or light colors.

Figure 3-1 illustrates the design conceptualization. It shows a privacy aware sharing application that is able to share data from various sources (both content and context), while using context input for privacy based decisions and providing feedback to the user.

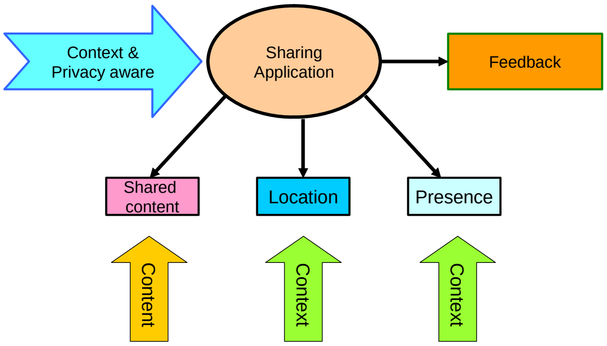


Figure 3-1: Conceptual design of the application

Based on this initial setting, several components that compose a privacy model were identified (see Figure 3-2). These components include the services, the applications, the control, the user and user's contacts. A service component has different pieces of software running in it. These functions include collecting contextual information about the house and its inhabitants, for example, recognizing the person in the environment, knowing their activities, controlling the security system in the environment. A control component allows a user to control the flow of data by setting his or her privacy preferences. A user's contacts can be, for example, a family member or a friend that interacts with the user. The arrows in Figure 5 show the data flow between components. Several privacy decision points are stamped between different components to control the flow of data.

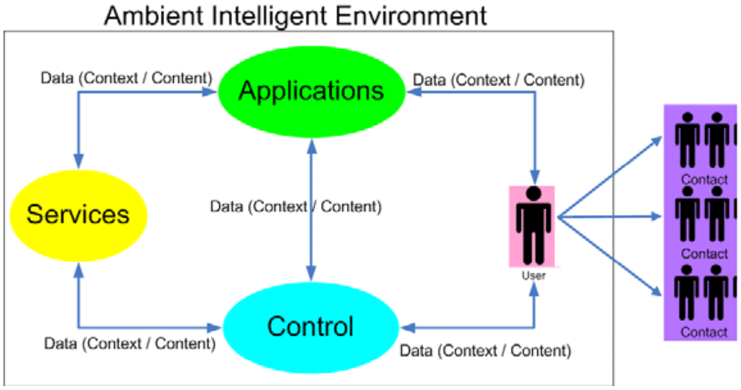


Figure 3-2: Components in the environment

A sharing application by which two people at different locations can share photos, location and presence, was chosen to implement the model. In addition, an interface for users to control the system by setting their privacy preferences was implemented. This sharing application is context dependent and reacts based on the user privacy setting. An existing system was used on which these functionalities could be added. An Amigo application was chosen for the implementation. This application, “Shared activities” met the capability of the design requirements and the design ideas. In “Shared Activities”, photos can be shared between remote places by using a TV as a display medium. The following section describes the unique interface used by this application and how it was adapted to the functional prototype.

3.3.1 Adapting “Shared activities” for prototype interface

In order to have a standard and consistent interface between different Amigo demonstrators, we decided to build our application on top of the Amigo application. The Shared Activities application user interface is built with CE-HTML and has many different menu options. The navigation complies with the EasyLogic 3.0 standard (Oosterholt et al., 2006, [1]). This standard, which is based on a “Three-Feet-Interface”, brings some unique requirements to our prototype. The principle of a “Three-Feet-Interface” is that a user should be able to control the interface by using a remote control while sitting comfortably on a couch. Due to this principle, the text displayed on the screen should be large enough to be legible. Besides, the way to navigate through the interface is limited to a few buttons on a remote control. The menu in the EasyLogic standard is divided into two columns, left and right. Left, right, up and down arrow keys are used to navigate through the menu. Four different buttons, red, green, yellow and blue are used for additional functions.

This interface was extended with a privacy management interface with which users can control the system by setting different privacy preferences. The main menu was reduced to only two main menus such as ‘Television’ and ‘Sharing Application’. Each of them has several submenus.

3.3.2 Sharing information

Presence, location, and photos of a user are shared in our application. Presence refers to a user status. This status has different values: ‘is online’, ‘is away’, ‘is busy’ or ‘is offline’. Location refers to where a person is located at specified point in time. Confirming evidence from the results of the user studies, the work of Sheikh et al. (2006), [2] and Lederer et.al. (2003), [3], shows that users should be able to choose the level of detail in which their location information can be shared. Furthermore, different sources of location information can have different levels of detail. In our application, these levels will be: “room”, “building”, “city”, for the physical location. Each of these levels can have the value “known location”, and “do not share”. The special case for ‘known location’ indicates how agreements between users can be used in combination with precision; this has been taken into account in our model. This model is depicted in Figure 3-3, in which AmbiLab is the room, WDC is the building, and Eindhoven is the city.

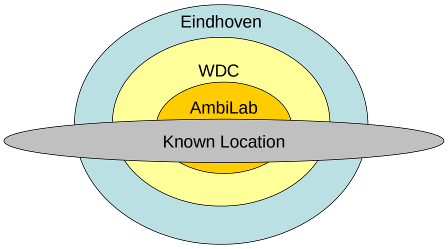


Figure 3-3: Example of different levels of precision for location

These different precision levels provide a user the flexibility to choose which information is to be shared. That is, they have control over the precision of their context information that will be disclosed to other people. The precision of the information determines also to what level the actual information is disclosed. Our model for the selection of location preferences is similar with Lederer et al.'s (2003) ordinal precision level. Instead of using their concept 'vague location', we use the value 'known location', in our application. PV

This value can be used as a custom location setting to simplify the preferences setting for the user. Furthermore, in our application a different precision level is shared depending on the relationship between the persons who are going to share the information. With regard to user control, the users can set their preferences, that is, 'who will see what and when'. The application will react according to these preferences setting. The users will configure their preference settings and maintain in this way the central control over their privacy policy. The privacy policy framework presented in Chapter 4 will expand on this concept.

3.3.3 Presenting context information

Presenting context information in an intuitive and lucid way is very important. Within the constraints of designing our application the use of icons, colored text and colored light was chosen. The photo-sharing application is presented by means of a photo icon, the presence status of a person by means of colored text, and the joined presence and activities status by means of colored light. With this selection we adhered to the constraints and opportunities provided by the "Shared Activities" application for of the user interactions.

Using colored light for presenting context provides ways to present context information without explicit interruption. It can support the exchange of information between two parties in a subtle way according to prior agreements and social conventions, while at the same time providing sufficient feedback for the application user. Four different colors were used to present different contexts. Two settings are used to present whether some one is logged into the sharing application or not. The other two settings are used to present whether other persons are or are not in the room with the users of the sharing application. With this presentation of context, the user located at one side gets feedback about the change in context at the other side, for example, when a child is leaving the room in our scenario. These settings represent a compound context. That is, the "context" that is represented in the example application is a combination of different information sources: activity, availability and location (of more than one person)

3.3.4 Amigo intelligent services

Since the prototype is part of the Amigo extended home environment, the Intelligent User Services from the Amigo middleware are used to retrieve contextual information and user profiles. The Amigo Context Management Service (CMS) is used to track the location of a user and store this location information on the server. The Amigo User Modeling and Profiling Service (UMPS) stores user profiles and can be used to reason on user context and feedback. These basic facilities provided by UMPS will be used and extended to store user privacy preference settings.

3.3.5 Privacy model for sharing applications

Crucial components for handling perceived privacy in the Amigo extended home environment were defined. These components are: the services, the application, the control, the users, and the users' contacts. The information flow between these 5 components results in a context dependent setting, i.e., how the application reacts is context dependent. This privacy model for sharing applications is shown in Figure 3-4.

Services X **Applications** X **Control** X **Users** X **Contact** = **Setting (Context Dependent)**

Figure 3-4: Privacy model for sharing applications

‘*Services*’ is a component to collect and provide contextual information about the environment and its users. The contextual information is stored in the centralized database and can be used by an application. This component can be any Amigo service. In our model the impact of a service is mostly determined by the sensitivity of the data that it provides.

‘*Application*’ is a component to provide a bridge between the technology and the users. Users can use an application to access contextual information. In our model, it is important what the application does with the data. As explained in the previous chapter it is expected that settings will differ per application and even per application usage.

‘Control’ is a component to provide a privacy preferences setting interface for a user to control the flow of information from one component to another component. It can be an independent interface or it can be embedded in an application. Page: 26 Furthermore, it can be possible that the user has not specified any control (conservative default) or has specified special cases (such as the absence of data) to be privacy sensitive.

‘User’ is a component to indicate that settings can differ for each person. This component is also influenced by context.

‘*Contact*’ refers to a user’s contact; it can be a user’s family member, a user’s friends or any other contact group defined by the user. It also has a context dependent factor.

Different combinations of these components create different settings.

The following example illustrates the role of these components in the privacy model for the scenario that was selected for our application:

if Maria [users] is in her room [services (context)] and

is using a sharing application [applications], and

her husband Jerry is alone [services (context)]

at another location [contacts] and

is also logged in to an application [applications],

then use setting A- share photos [setting (context dependent)].

Setting A – share photo - is a setting that is preset by Maria by using the privacy preferences setting interface. In this example, Maria and Jerry are at different locations and there are two different services and applications running at the same time at these two different locations. Setting A is a decision made at Maria’s location to determine the dissemination of the information (in this case, to share photos). Table 3-1 shows the extension of this example of the privacy model.

3.3.6 Applying the models

It is assumed that all the needed Amigo Intelligent User Services, in particular, CMS and UMPS are installed in the two Amigo homes. The application that will be the intermediary for all services is the sharing application. This application allows users at different locations to share information and activities, provided that they are Amigo home users. Users can control the application by setting their privacy preferences from the application privacy management interface. A simple form of identity verification is required. It was decided that users would use a personal identification number (PIN). It was also decided to not yet use the Security provided by Amigo at this point but rather to assume that this is present. Furthermore, services that are already embedded in the home can be accessed and used by the application.

Table 3-1: Example of applying the privacy model in an Amigo home

Services **(Context)**	**Applications**	**Control**	**Users**	**Contacts**	**Setting (Context dependent)**	
					Context	**Enforced Privacy Policies**
Location: In room At home Not at home	*Sharing content*: Photo	*Interface for privacy preference setting*	*Any user*	*User's contacts:* Friends Family Colleagues	Person A: at home	Setting A: Share
					Person B: at home	Setting B: Do not disturb
					Person B: is alone	Setting C: Accept sharing request
						Setting D: Do not share

The services in the homes are activated in three different stages: login, after login and log out. Once the user logs in to the application by selecting his or her name and pin, the application first authenticates the user's password by communicating with UMPS. If the pin is valid, the user can start using the application to share information and activities with his contacts. The information that is being shared consists of user location, user presence and user content (photos). The activities being shared are browsing photos together. A lighting service will set the room light according to the privacy setting and will change dynamically if the context of the user is changed. When the user initiates a request to his or her contact to share an activity, for example, to share photos, the contact will receive a notification. If the contact accepts the request, the photo sharing session is started. A list of photos from the user is displayed on the screen; the photos are arranged in private and public categories. Only photos in the public are to be shared. The photo sharing session can be ended at any time during the sharing session. A user can modify his or her privacy preferences at anytime after login to the sharing application, for example, with whom he or she wants to share the photos; which location precision level to be shared with his or her contacts. The application takes care of the privacy settings of the user. To end the sharing application, a user logs out from the application. A sequence diagram to login to the application is shown in Figure 3-5. In our example, the privacy settings are retrieved once by the application when it is started and then enforced internally, which is why little communication with Amigo services takes place. A future improvement would be to react to change events from UMPS (not shown in the sequence diagram). Figure 3-6 shows the sequence diagram when a sharing request is initiated and photos are being shared.

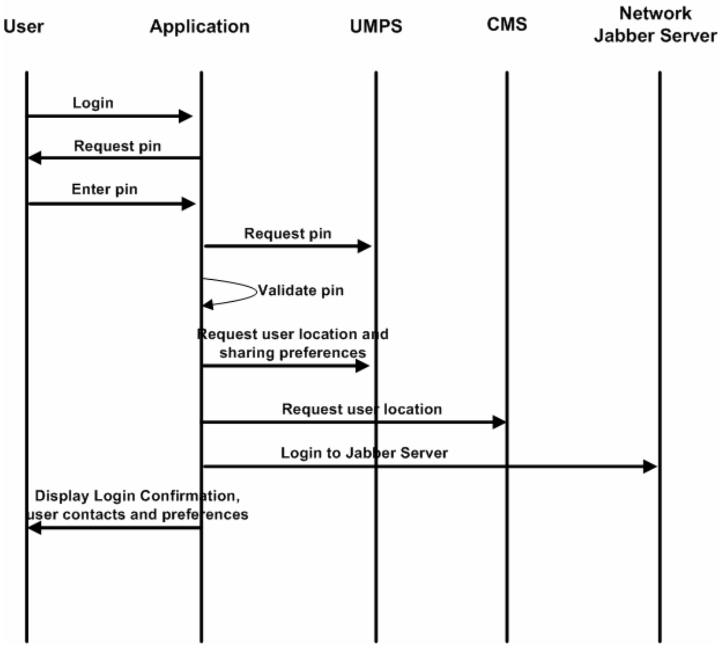


Figure 3-5: Sequence diagram for login to the application

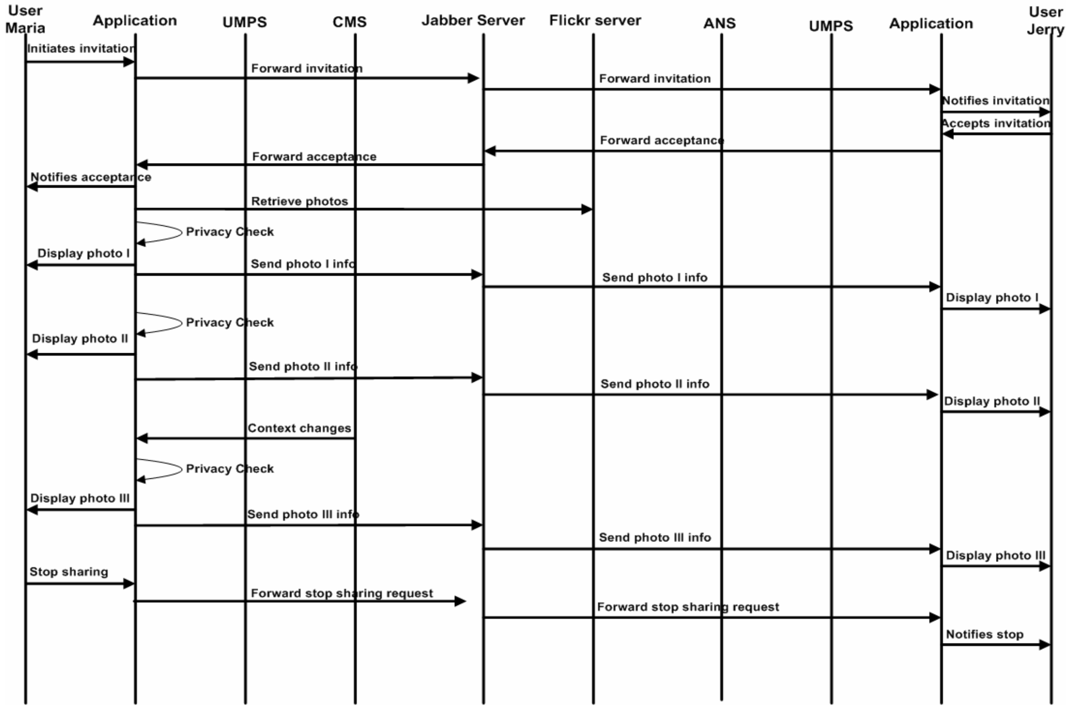


Figure 3.6: Sequence diagram for initiating a photo sharing request

3.4 Implementation

In this paragraph the technical implementation and architecture of the prototype application are described. The different components in the architecture, including the Amigo server, the Amigo client and the network to connect different homes are outlined.

3.4.1 Abstract application architecture

It is assumed that different services and applications are installed in two different Amigo homes. There are also different ways to connect these homes. Figure 3-7 shows how we connected these two homes in an abstract view. Typically, there are three main components in each Amigo Home, the Amigo server, the Amigo client and the Internet. The server consists of different services running in the home. It acts as a centralized database that stores contextual information about the home and its inhabitants. The client can be any display device that is connected to the Amigo server and the Internet. An application is installed in an Amigo client. One Amigo client can have more than one application running in it. Context information is shared between two Amigo homes through an application and the Internet. A user is given control over which information is to be shared. The Jabber server is used to store the user's contacts information. The Flickr Server [4] is used to store the user's photos. Several privacy decision points are stamped at different locations to control the data flow from one component to another component.

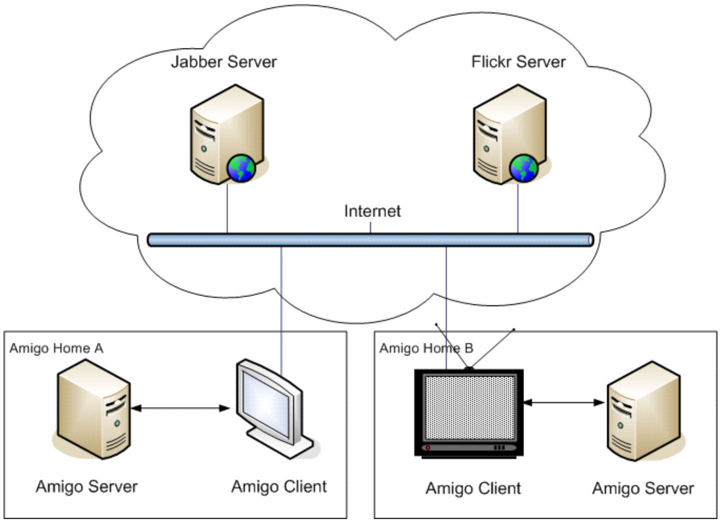


Figure 3-7: Abstract application architecture for two Amigo homes

3.4.2 Application architecture

In the prototype, information is shared between the Amigo server which is running various Amigo Intelligent User Services, the Amigo client, which is running the privacy aware applications, and between different homes. Before information is passed from one component to another, it will first go through a privacy decision point. Privacy Decision Point (PDP I in Figure 3-8) is used to control the information that is being shared between the Amigo server and the Amigo client. PDP II and PDP III in Figure 3-8 are used to control the information that is being shared between different applications across different homes, going out from Home A and into Home B respectively. Figure 3-8 shows the application architecture with the PDPs located at different stamp points.

The PDP consists of rule-based software that reacts based on user privacy preferences. User preferences are stored in UMPS using the XML format (for details see UMPS manual. Note that a refinement is presented in Chapter 4). Once a requester (for example, an application) requests information, the software will check the user's privacy preferences. The information that passes the rules is sent to the requester. Besides different PDPs, the application architecture shows the three main components in the prototype: i) Amigo Server, ii) Amigo Client, and iii) Network. The Amigo Server consists of different services that are running in an Amigo home. They provide different types of information to the client. The Amigo Client consists of different applications that can be used in an Amigo home. These applications are, for example, the sharing application, a lighting application (Living Light), and a game application. The Amigo Server contains a configuration server, the UMPS and the CMS. The Network is used to allow more than one home to be connected. In the prototype demo, an Internet connection is used to establish the connection between two computers. Each computer is connected to a television.

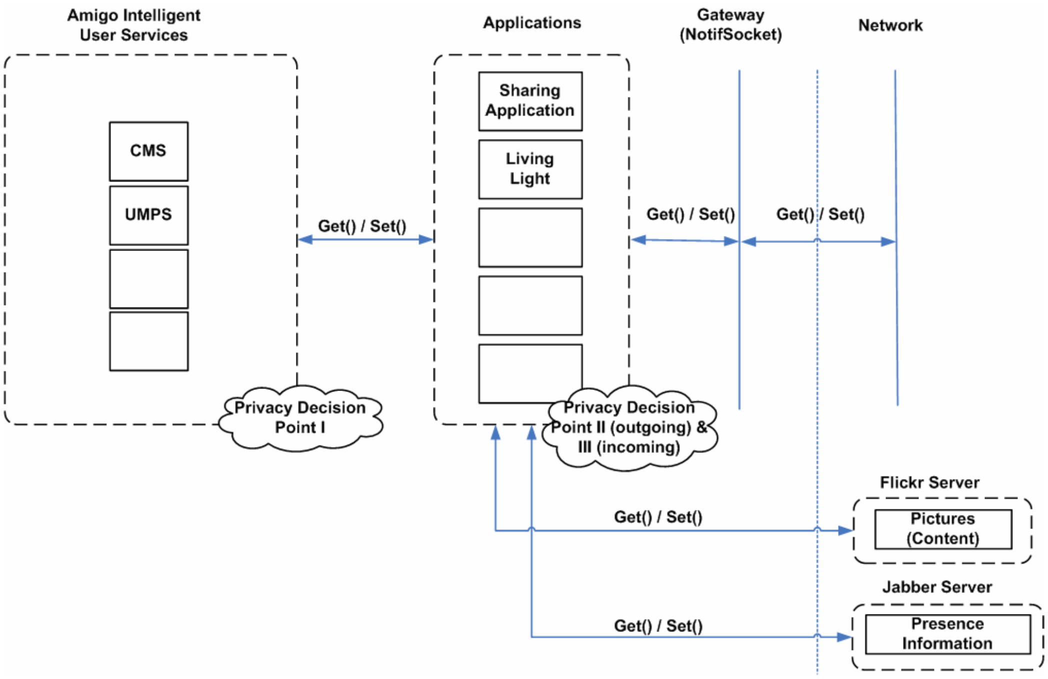


Figure 3-8: Application architecture

By using the application, users can exchange information, share activities, play games, chat, feel each other's presence and share their social context. The main functions in this application are sharing information and activity in two or more Amigo homes. In order for this to work and taking into account the privacy issues, several functions are implemented:

1) Login / logout from the application

2) Sharing information – presence context, location and application status

3) Sharing activity – share photos with people at other location

4) Setting privacy preferences

The application is installed in a CE-HTML enabled television. Users can share different information with different people that are located anywhere in the world at the same time. They have the freedom to choose which information they want to share and with whom, by means of the privacy preferences interface in the application. The three sources of information are provided by, i) the Amigo service, for example, the location of the user, ii) an application, for example, the status of the application, i.e., on or off, and iii) the user, for example, the availability, i.e., online, busy, away or offline. A default setting is assigned to each new user. They can modify their privacy settings from the privacy preferences interface. The PDP is enforced to make sure that the application will react according to the privacy setting and the user context. Users receive the feedback from the application interface. Different user states are presented by different colors of light, e.g., online/offline/sharing photos. A LivingColors[3] lighting system used for this purpose.

A user can share photos with his or her contacts by using the application. Before sharing can occur, the photos should be uploaded to a server. Each photo that is uploaded to the server is tagged with a different tag ID. The ID can be ‘private’, ‘public’ or anything else depending on the need of the user. Once a user invites a contact to share his or her photos, PDP II is enforced to select the appropriate set of photos for sharing. The selection is based on user photo privacy preferences, taking into account the tag ID of the photos, the presence and location of the user. The set of photos can change accordingly when the context of the user changes. Users can set their application and location privacy preferences using the interfaces from the application. They choose whether they share their application with someone or not. This implies that if they agree to share their application with X, only then can X share the application with them. They also choose whether to share their location information with someone or not, and at which level of detail this will be.

The Network allows the connection between multiple homes. In our prototype, the application communicates via a Jabber server (Jabber Software Foundation, 2006 [4]). A Flickr server is used to store the photos. The Jabber Server is used to exchange messages and user availability between two Amigo homes. The Extensible Messaging and Presence Protocol (XMPP) are used to allow this to happen. The types of messages exchanged include the user’s location information and application activity. The user’s availability is shown as the user’s current status, i.e., is online, is away, is busy or is offline. The Jabber Server is also used for initialization of a photo sharing request and establishing the connection between the two homes. The Flickr server is used for uploading and storing the user’s photos. Each photo is tagged to indicate the ‘privacy level’ of that photo.

3.5 Expert walkthrough of the application

The prototype was demonstrated to expert users to generate feedback and to validate with regard to the initial user requirements and design guidelines. The demonstration was set-up as a walkthrough based on the initial usage scenario. The prototype was set up in the Ambi-lab at Philips Research. Two display devices were used to show the participants what is happening in Maria and Jerry’s location. The test room was organized in such a way that it would represent two homes (see Figure 3-9). The functionalities of the prototype were shown and discussed. One of the functions of the system was that it allows a user to share his or her information (for example, user’s photo, user’s location and user’s presence) with his or her contacts. Another function of the system was that it detects the location of a person in the house. Participants were also shown an example of how to change privacy preferences settings by using the privacy management interface. For example, to allow users to decide with whom they want to share their actual detailed (room) location. By having all the privacy

[3] ©Philips

preferences settings in the database, the system can select the appropriate set of photos for sharing based on these privacy preferences setting. And, it can change the set of photos to be shared accordingly when the context of the user changes. In the same way, the system selects the color of the light and changes this when the context of the user(s) changes. The choice and meaning of the color is based on a social agreement between the participating users.

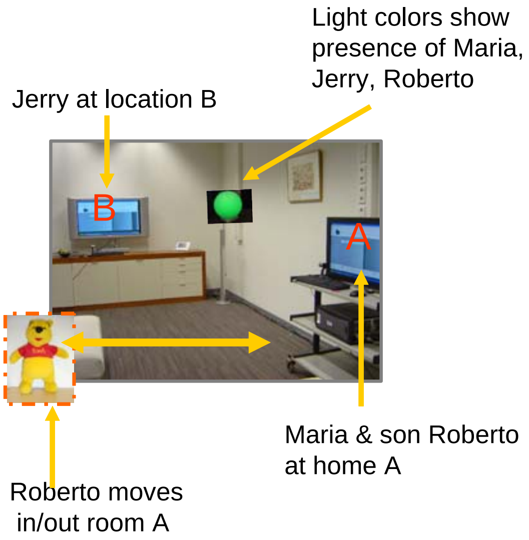

Light colors show presence of Maria, Jerry, Roberto
Jerry at location B
B
A
Maria & son Roberto at home A
Roberto moves in/out room A

Figure 3-9: Laboratory setting for expert walkthrough

At the end of the demonstration, the experts (n=10) were asked their feedback in an informal fashion. This feedback was then structured according to the design guidelines that were derived from the user studies.

Design Guideline #1: *Provide proper security to prevent misuse of information and inform users of security measures*. The prototype can be used by different family members. They each have their security PIN. Their preferences are associated with their identity. These data can be private and sensitive. Other people cannot view or access their personal settings without their consent. The user interface and the way of entering the settings are amenable to improvement and/or other user interaction means.

Design Guideline #2: *Provide control to users over privacy settings at several levels of increasing complexity.* The prototype provides control over the privacy settings at several levels. This control is implemented by allowing users to set different privacy preferences from the privacy management interface. By using this interface, they have several options for sharing and controlling: ‘Sharing Application Setting’, ‘Location Preference Setting’ and ‘Photo Preference Setting’. They can also deny or accept incoming requests.

Design Guideline #3: *Present the user with a choice of level of detail in which the information should be shared.* Each type of information in the application can be shared at different levels of detail. Users can adapt this level of detail to the context in which the information is shared. For example, they have different options to disclose their location information. They can either let other people know their accurate location or higher-level location information depending on their preference and the relation with other people.

Design Guideline #4: *Provide clear feedback over shared information*. The prototype implementation shows the user’s information that is being shared with the other party on screen. It shows what the person is sharing, with whom and at which level of detail. The feedback is real-time and consistent with the EasyLogic user interface style guide. For example, the screen shows that Maria is online, that she has her photo sharing application on and that she is sharing her room location with Jerry. At the same time, the same screen also shows that Jerry is online, that he has his photo sharing application on and that he is sharing his city location with Maria. Since Maria's name is highlighted, it is understood that the information is shared between Maria and Jerry. Moreover, Jerry is the current user of the application while Maria is one of his contacts.

Design Guideline #5: *Ask for user consent before sharing information*. The prototype implementation provided a confirmation message asking whether someone will be added in the shared list or not.

Design Guideline #6: *Avoid using automatic intervention to maintain user privacy.* This guideline initially assumed a 'don't automatically stop the application without user consent'. The prototype is context-aware and adapts to changes in the user's context depending on the privacy settings. For example, if someone enters Maria's room, the photo set that is being shared at that moment, will be changed automatically into another photo set and the change in context is presented by a change of light color. In this case the sharing application is not automatically stopped but changed to another mode without explicit consent of the user. That is, if the system is context aware and the user's privacy preferences are configured correctly, it might be possible to preserve user privacy without causing undesirable social situations, i.e., in narrowly pre-defined context settings.

In summary, the feedback from the expert walkthrough resulted in refinements for the design guidelines. These refinements are:

- Provide usable security and inform users of security measure
- Provide users with automatic and /or manual choice of level of detail in which the information should be shared
- Provide clear feedback over shared information by audio, video, text or images or in combination of any of these
- Avoid using automatic intervention to maintain user privacy if it is not preset or preconfigured by the user

Other points that were raised by the experts concerned trust, system autonomy, presentation of context information, and preferences for information disclosure. These points will be addressed in the discussion section.

3.6 Discussion

A running prototype system provides a much better carrier for eliciting user feedback with regard to privacy considerations than design concepts and visualizations. Such a system has the advantages of demonstrating the consequences of changes in context and can support the exploration of different situations. It also has the disadvantages of realistic system implementations as they always will carry a history of previous design decisions and business rationales.

The experts that participated in the walkthrough had very different backgrounds, among others, user system interaction; security and encryption; software engineering; system architecture; and CE applications). They trusted the system. They trusted it even more than they trusted other users. One of their quotes: "If I know the capability of the system I will trust it; I will have less trust in the people who will be using it". It was also stated that the system "will be trustworthy if it takes a conservative decision". This implies that the default value for the privacy setting should be very rigid. It is not desirable that other people know what these settings are. They also wanted a central control or setting to control the application themselves. Another issue regarding trust was expressed as: “Can you trust the system that it is sufficiently aware of the context to adapt automatically to the right setting?” Being *sufficiently aware* is the crux that needs more exploration and research work.

The relation between system autonomy and privacy is another critical factor. Being able to cheat was one of the requirements revealed by the user studies. Setting different privacy preferences and changing them during a session elicited discussions on how this might affect people’s social relationships. With regard to the level of detail for the preference settings it was clear that different levels have different meanings for different people. The following verbalizations illustrate these points:

- “People don’t need to know that you have set some kind of privacy setting for them”
- “If I change my privacy setting for a particular friend, people will infer something, as if I wish to hide something”
- “If I really wanted to show my privacy settings, I should be able to override them, by having a kind of ‘don’t embarrass-me’ button”.

Presenting the context information by means of colored light provides opportunities for personal and intimate ambiance sharing. For this the system should be flexible. The prototype only used one light per room, but participants actually desired to have a light presentation for each individual person.

The different levels of precision that were maintained in the application for the disclosure of personal information needs further exploration. It is quite clear that people require control over their personal information and that they like to vary this level of detail depending on the context at hand. These levels are different for each individual. It appears that people require about 3 levels of detail, but that these levels are at different depth for each individual. These settings might potentially cause social embarrassments and conflicts.

In sum, the application implementation made it possible to explore and experiment with privacy settings in a context-aware environment. The results highlight the need for more exploration in different settings and conditions. But it also identified that some privacy controls should be handled in a generic fashion, such as employing privacy rules. This approach will be presented in the next chapter

4 CAPriCE - Context Aware Privacy preferences Collection and Enforcement

4.1 Introduction

The previous chapter derived design rules for application developers on how to deal with privacy (preferences) of users. Next to design rules for applications there is also a more generic aspect for context ware environments on how to deal with privacy with respect to context that is typically measured or derived in such environments.

Context-awareness envisions services to have the ability to access real-time information about their users' environment and use it to customize service provisioning. This raises other issues of privacy enforcement. First and foremost, context information will not be entered by the user at a console, but is collected through ubiquitous sensors and software components without always notifying the user. Secondly, whether a user (context owner) will allow a particular requester access to some context information about him in a particular instance depends on the current context, for example: let my boss know my location only when I am at work or during office hours. This requires considerable extension to the expressiveness of current policy standards. Another problem that arises is the complexity of authoring such policies exposed to end-users, especially novices. For users to accept context-aware systems, they will, on the one hand, want fine-grained control on who get access to their context information and under what conditions while, on the other hand, they are unwilling to fill in very detailed context access policies.

CAPriCE is a privacy policy framework that is well-suited for context-aware systems; its architectural highlights are discussed in section 4.5. A model for an end-user privacy preference enumerating its necessary components is presented in section 4.4. The extensible features of XACML are used to represent context-aware privacy policies and application templates; extensions were made to an XACML PDP to evaluate these policies.

An additional feature is that the CAPriCE privacy policy framework allows application developers to author 'policy templates' which act as the default privacy configuration for their users. By using templates, an application specific default is provided, which makes it clear what context is requested by the application, and thus what the relevant privacy policies are. An additional benefit is that new policies are formed only if and when users customize templates. As most - but especially novice - users tend to use defaults, this means that the number of policies is essentially determined by the number of context aware applications and not by the number of users, addressing scalability issues that may arise when only per user policies are used.

4.2 Related work

Wishart et al. [1] propose a system in which privacy policies are expressed in two distinct levels:

- A privacy preference that specifies whether a subject has access to a type of context.
- A granularity preference that specifies the maximum allowable quality of context to that subject.

They have proposed a simple rule language to encode these preferences but do not specify how conflicts would be resolved. For example, Maria allows Jerry to know her location when she is working but not when she is at home. Will Jerry gain access to her location when she is working from home? Policy standards such as XACML tackle such problems effectively [1]. The reason for bifurcating policies into these two levels is not sufficiently motivated, especially considering the fact that it would double the already high performance overhead of policy evaluation with each context request. In our opinion, end-user privacy preferences to control access to their information (Boolean) and to control the provided quality of context are not that different.

Hull et al. [2] also propose a privacy policy framework system in which users are classified according to their role, e.g. office worker, housewife, and system administrators can define privacy templates for each user class. This approach is quite comparable to ours with some differences. Firstly, we distribute the responsibility of authoring privacy policies and templates according to the context-aware application in question. We think that the knowledge of application developers about all aspects of the application they are developing is paramount in setting the baseline for privacy preferences for users of their application. Also, the customization activities that users perform for each application, including their operating system and firewall etc., are disjoint, which gives rise to application-based thinking. We allow users to set context-aware privacy preferences following this model by giving them access to the application 'defaults' set by the developer and customize them according to their wishes. Secondly, Hull et al. lack the notion of quality of context in their policy framework. Thirdly, they do not make clear how policy conflicts will be resolved, which we address by extending the evaluation techniques provided in XACML.

Tentori et al. [3] identify a set of 'Quality of Privacy' variables that need to be controlled by the context owner, in this case hospital staff, to protect their privacy in pervasive systems. They propose an ECA (Event-Condition-Action) based messaging system in which the event contains the situation of the context owner and the time of context request, condition contains information about the precision of context that the owner is willing to share with the requester and action represents any functions that might need to be executed to obfuscate context information. Although the XMPP based messaging system that they describe in [3] might be useful, their approach for encoding privacy preferences is much more restrictive than ours. They only consider a small subset of context variables pertaining only to the context owner to decide on what action to take. In our case, the precondition might contain information about the owner, requester or any third party. Their privacy preferences do specify the max allowed QoC but there is no system of conflict resolution, i.e. computing the resulting permissions when two applicable preferences allow different levels of quality. The indispensable issue of providing the end-user with defaults to ease policy authoring has not been considered altogether.

4.3 Privacy Enforcement in context aware environments

CAPriCE builds on the concepts presented in [4] about Quality of Context (QoC). In the interest of readability, we will describe these concepts in short over here. Sheikh et al. [4] present five QoC indicators that describe the privacy sensitiveness of context information and also specify techniques to quantify them in absolute terms. These indicators are as follows,

1) *Precision* represents the granularity with which context information describes a real world situation. E.g. a doctor requires a patients' body temperature with at least three significant figures precision (such as 36.3°C).

2) *Freshness* denotes the time that elapses between the collection of context information and its delivery to a requester. E.g. a doctor requires that a patients' body temperature is not older than 1 hour.

3) *Temporal resolution* signifies the period of time to which a single instance of context information is applicable. This might vary due to the sampling rate of the context source. E.g. the temperature of a room collected every 8 hours is valid for a period of 8 hours after it is collected.

4) *Spatial resolution* symbolizes the precision with which the physical area, to which an instance of context information is applicable, is expressed. E.g. a building security system that keeps track of the number of people present in the building may provide this information with the spatial resolution of a room, a floor, a section of the building or the whole building.

5) *Probability of Correctness* corresponds to the probability that an instance of context information accurately represents the corresponding real world situation, as assessed by the context source.

Although all these indicators are useful to protect privacy, for reasons of brevity and simplicity we will consider only one indicator, i.e. Precision. We use the context type *'activity'* to demonstrate the use of QoC in privacy policies. Table 4-1 shows four levels at which the activity of a user can be described, irrespective of how it has been determined.

Table 4-1: Precision levels for context type 'activity'.

Precision	Activity Information
0	No access/unknown
1	Available for telephone call (yes/no)
2	High level activity. Possible values: work, home, sport, travel-work, travel-personal
3	Exact activity. Possible values: watching TV, in a meeting, browsing internet etc.

4.3.1 Motivational example

To demonstrate how user privacy can be protected using the proposed QoC indicators, we describe a real-world scenario where we will use the quantified QoC parameters described above to create privacy policies through which users can specify the QoC that may to be provided to requesters in different situations.

Telemonitoring is a process in which different physiological variables of patients, who require constant medical supervision, are measured and assessed while the patient is not present in a hospital. The AWARENESS [5] project illustrates the scenario of an epileptic patient who requires constant telemonitoring due to the risk of sudden seizures. A patient experiencing an epileptic seizure may lose all control and start shaking rapidly, and immediately needs attention of another person who can move him or her away from dangerous objects and provide immediate care [7]. In the AWARENESS telemonitoring scenario, patients wear a set of sensors that collect real-time health information and forward this to a PDA for processing. These devices are collectively known as the Body Area Network (BAN) [6]. The health signals collected get assessed in and by the PDA and may be communicated to another location for more processing or viewing by a health care professional.

In our scenario, Maria is a patient of Dr. Bernard who monitors Maria's health condition using his Health Monitoring System (HMS). Voluntary health care providers such as Maria's family members and friends are registered and may be contacted by Dr. Bernard, using the HMS, if Maria is in a medical emergency and needs care. The application developer of the HMS, using his understanding, makes the following privacy policy for Maria and her care givers:

> *If there is no emergency, allow Maria's doctor to see Maria's activity at level 3. Allow Maria's doctor to see activity at level 2 of all care givers. The doctor must be using the HMS application.*
>
> *If Maria is in a medical emergency, allow all medical professionals using HMS to see Maria's and her care-givers' activity at level 3.*

Now, Maria is comfortable with whatever information the HMS might provide to medical professionals as she is a chronic seizure patient. Her caregiver and husband, Jerry, on the other hand is not comfortable with his context information being provided to Maria's doctor(s) at all times, especially when there is no medical emergency. Therefore, he wants his privacy policy to look like the following:

> *If Maria is not in an emergency, then Maria's doctor can only see whether Jerry can be disturbed (level 1) only when (s)he asks for this information through the HMS.*
>
> *If Maria is in an emergency, let any doctor know my activity at level 3.*

In the next section the components of these kinds of privacy preferences are discussed.

4.4 Structure of users' privacy preferences

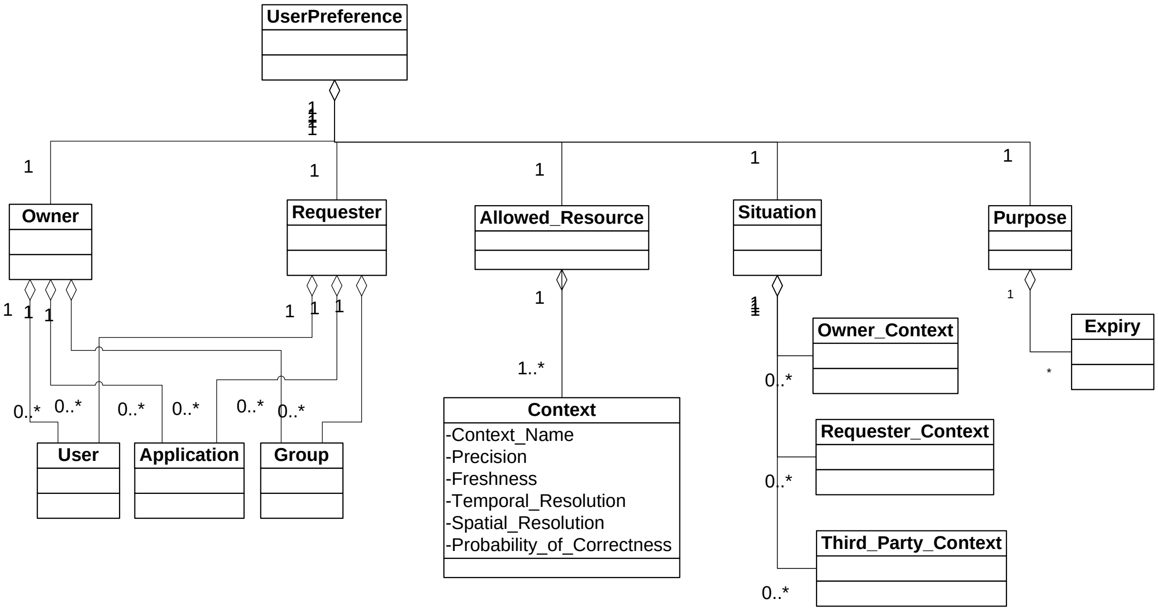


Figure 4-1: UML representation of User Privacy Preferences.

After studying different privacy and access control policy standards ([8][9], [10]) and our experience with context-aware systems, five main components of a context-aware user privacy preference were identified, shown in Figure 4-1. A description of these components is as follows:

Owner: This is the human user about whom information may be shared with other parties. No context information can be shared unless this user has either authored or consented to a privacy policy explicitly authorizing the communication of a context type in question to a certain requester. In the example this is Maria or Jerry.

Requester: The party requesting information about a context owner. This may be another human user or a context-aware application, or both. In the example this would be Maria's doctor or any medical professional.

Allowed resource: This is the information which this preference refers to. It consists of the allowed context type and the corresponding allowed QoC, such as activity at level 3 from the example.

Situation: This part is specific to context-aware privacy preferences and denotes a set of context variables that represent a real-world situation. Using this part, a context owner can specify values of context variables, which represent a real-world situation, to control access to his privacy sensitive information, an example would be ‘Maria is in an emergency’.

Purpose: In this part of a privacy preference the context owner can provide the allowed set of purposes for which a certain type of information might be requested. Thus, the request would be granted only if the purpose in the request matches the one in the privacy preference. Such a ‘purpose’ is a customary constituent of most privacy policy standards. An example would be ‘Health monitoring’.

In the scenario presented in the previous section, Jerry is a care-giver to Maria and wants to protect his privacy by authoring a more restrictive policy than what was presented to him. In Figure 4-2 we show Jerry’s more restrictive privacy policy mapped onto the UML structure of a user privacy preference as presented in Figure 4-1.

The preference limits the precision of the location information to level 3, while the activity precision is limited to 1 in case Jerry is in the same city as Maria and she currently does not have a seizure (Emergency(Maria)=False).

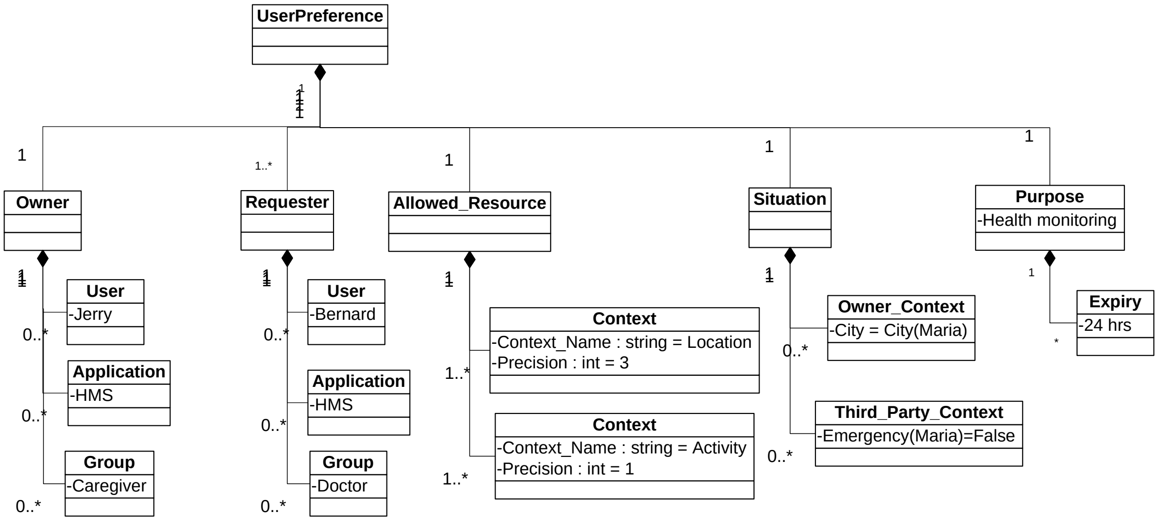


Figure 4-2: Caregiver's personalised preference.

4.5 Architecture highlights

In this section we describe the architecture of the privacy policy framework called CAPriCE (Context Aware Privacy preferences Collection and Enforcement). A high-level depiction of the architecture is shown in Figure 4-2. For describing privacy policies of a context owner, the XACML privacy profile [1] language was chosen.

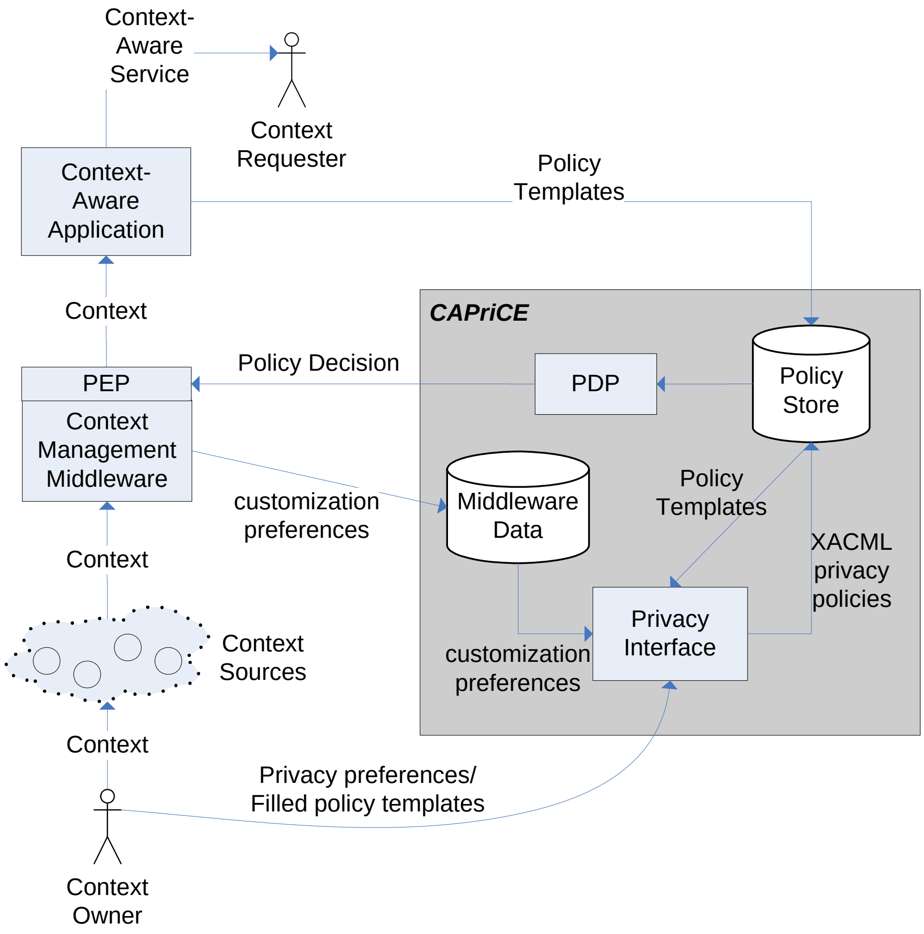


Figure 4-3: CAPriCE Architecture.

The main components that play a role in the CAPriCE framework are:

Middleware data store: The information required from the middleware, that would help context owners to author privacy policies, is stored in a relational database.

Privacy Interface: comprises of a set of web-pages that give dynamic options to the context-owner to author policies based on information, such as what context types about the user requesters can access, what quality levels they can see, through which applications etc.

Policy store: A database for storing (XACML) policies. For the example the native XML database from Apache called XINDICE [12] is used. This facilitates querying specific parts of policies for reconstruction of the privacy interface.

PDP (Policy Decision Point): is the component responsible for accessing the XACML policies relevant to a request, evaluating them, and providing an access decision for the PEP to enforce.

PEP (Policy Enforcement Point): The middleware implements an XACML Policy Enforcement Point (PEP) which means that it sends a decision request to an XACML Policy Decision Point (PDP) before providing context to any requesting party. The PEP receives an access decision from the PDP that contains the Quality of Context (QoC) level that the middleware is allowed to pass on to the requester.

Furthermore there are also components that, although they are external to CAPriCE, still play a vital role in its operation.

Users: In a typical context exchange in a context aware system there are two types of users involved. The context owner is the user about whom context is collected; the context requester is the user to which the context is provided. The users may be only indirectly involved, e.g. via an application that they are using.

Context aware application: The requester is, typically, using a context aware application that provides a rich service to the requester using context information that it has collected from the context management middleware. It is interesting to note that the context owner might use services that use his own context in which case he becomes the requester himself. Context may also be requested by someone else, more than one person or organizations that cannot be classified as a human user. For the immediate discussion the precise form of the requestor does not play a role, a complete taxonomy of these situations is also out of scope of this deliverable.

4.5.1 Context management middleware.

The part of context management middleware specific to CAPriCE works as follows: The middleware populates its customization preferences into the middleware data component at the start. Then each context-aware application that wants to be used by users registers with CAPriCE during which it stores its policy templates in the policy store. When a context owner wants others to gain access to his context through one of the registered applications, he is shown the templates registered by the application as a set of options on the privacy interface. The user can customize these or add own privacy policies for this application.

A requester uses this application to access the context owner’s context and the application requests the required context from the middleware. The PEP in the middleware asks for an access decision from the PDP before providing context to the requester. Note that this means that the PEP is tightly bound to the actual context infrastructure used, much more so than the PDP. The PDP evaluates the context-aware privacy policies given by the context owner. The PDP procures any additional context parameters that may be required for policy evaluation directly from the middleware. Notably, context requests from the PDP are not subject to privacy policy evaluations. Note that this may indirectly reveal privacy sensitive information, since the decision to grant access to context at a certain level may be determined by context itself. The outcome of this decision process may therefore indirectly reveal context information that it is based on. The risk on this is however not great, while avoiding these risks would add considerable complexity to the CAPriCE framework. It was therefore decided to leave this issue as is.

The PDP, after policy evaluation, provides an access result to the PEP within the middleware which contains a maximum allowable QoC level. The middleware then makes the requested context type available to the requester at the allowed QoC level. It is the responsibility of the context middleware to potentially lower the QoC of a piece of context information to the allowed level, since the PDP only knows about generic (integer) levels for allowed QoC. Of course it may be the case that the raw context in itself is already of a lower QoC level than is allowed, in which case no changes to the context are needed.

4.5.2 Privacy policies in Amigo

XACML (eXtensible Access Control Markup Language) [1] is used to encode and communicate end-users’ privacy policies in AMIGO. XACML is an OASIS standard and is widely adopted in research and industry. SUN Microsystems has developed a set of XACML policy evaluation libraries [11] that have been used to evaluate privacy policies within the AMIGO Context Management Service.

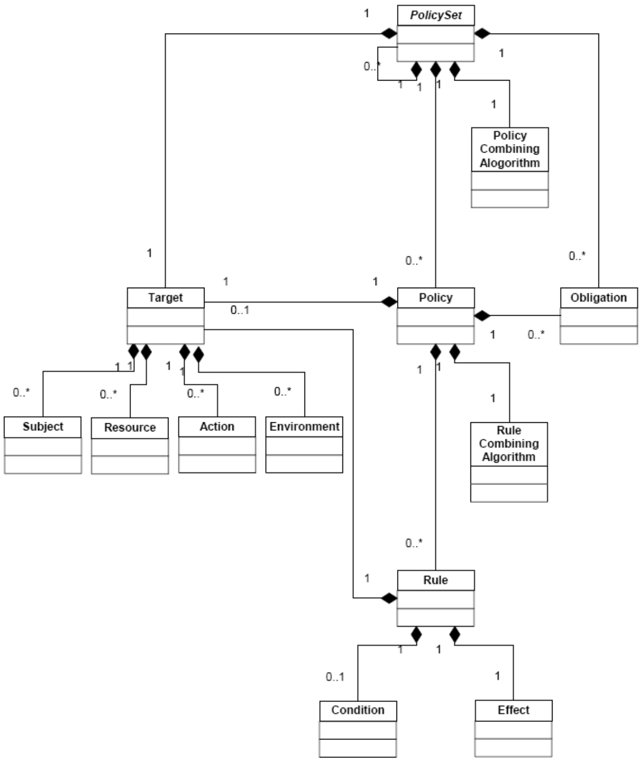


Figure 4-4: XACML policy language model.

Figure 4-4 shows the language model of XACML policies.

XACML, like other access control policy standards, treats a resource as an atomic entity. Thus, all policies provide a Boolean reply after evaluation, i.e. whether a particular access request should be granted or not. However, as shown earlier, to provide effective privacy control to end-users, they should be given control over the quality of context (QoC) information that is provided to others [4]. Therefore, the policy evaluation libraries were extended so that users can define privacy policies that specify a maximum allowable QoC (e.g. location information only at city level) in a particular situation (e.g. depending on day of time and the identity of the requester).

The maximum allowable QoC is expressed as an integer, which may have a different interpretation depending on the type of context to which it refers, but with higher numbers always meaning higher precision. This was done to keep the policy related parts of the architecture generic. If the results of an evaluation would depend on the type of context to which it refers, the PDP would have to be ‘context aware’. By keeping the result generic, in the form of an integer result, the PDP does not have to be context aware and is able to produce results independent of the type of context to which the policy refers. It is the responsibility of the context middleware to translate the allowed levels of QoC to the specific context types, in other words: the PEP is context middleware specific.

4.5.3 Mapping of Quality of Context to the Amigo Ontology

The Amigo vocabulary defines the concept of a ContextParameter. This data is used for registration and discovery of specific context via the context broker. Examples are UserLocation or Bluetoothscan. In this concept there is a possibility to add a precision attribute, which is how the Amigo context links to the privacy mechanism (please refer to the CMS section of this document, which describes this in detail).

As an example, the data below describes the location of User ‘Jerry’ in two different ContextExpressions with different precision. Note that the representation is not necessarily valid XML or RDF, since it is simplified to make the mechanism more clear.

```
<rdf:RDF>
<contexttransport:UserLocation>
        <amigo:TimeStamp>29-11-2007 11:35:48</amigo:TimeStamp>
        <amigo:hasSource rdf:resource="#LocationService" />
        <amigo:Nature>amigo:dynamicnature</amigo:Nature>
        <amigo:Precision>4</amigo:Precision>
        <amigo:isLocationOf rdf:resource="#Jerry" />
        <amigo:isLocatedIn rdf:resource="#Kitchen" />
</contexttransport:UserLocation>
<contexttransport:UserLocation>
        <amigo:TimeStamp>29-11-2007 11:35:48</amigo:TimeStamp>
        <amigo:hasSource rdf:resource="#LocationService" />
        <amigo:Nature>amigo:dynamicnature</amigo:Nature>
        <amigo:Precision>3</amigo:Precision>
        <amigo:isLocationOf rdf:resource="#Jerry" />
        <amigo:isLocatedIn rdf:resource="#Home" />
</contexttransport:UserLocation>
<amigo:User rdf:about="http://amigo.gforge.inria.fr/owl/RFLocation#Jerry">
        <amigo:identifier>Jerry@JGAV300600.amigo.net</amigo:identifier>
        <amigo:isLocatedIn rdf:resource="#OutsideTheHome" />
</amigo:User>
```

The precision of the context is expressed as a number, where a higher number means a higher precision. In the example above, room level precision has level 4, whereas building level precision has level 3. Note that although the precision is expressed as a number (which is better suited for processing), the meaning of the number is specific to the type of context it refers to. In this way, it is possible to apply QoC on the CMS ontologies and thus allow PDPs to select the appropriate context information (for example by adding it into the SPARQL query).

4.5.4 PDP extensions

Several extension have been made to the SUN XACML PDP implementation to process context-aware policies.

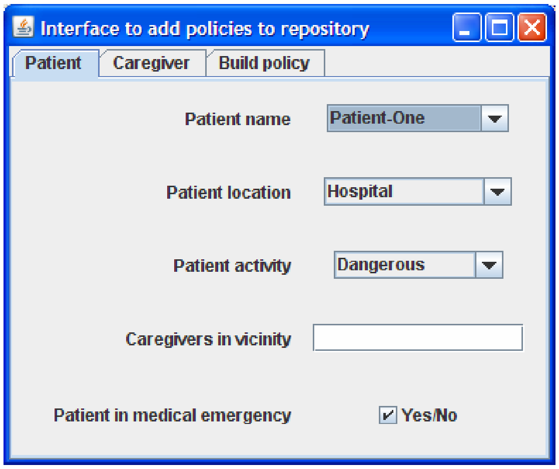
- A policy finder module to find all policies for the application being used by the requester(s) is added. It also finds all personalized policies of the context owner(s) applicable to the specific situation.
- The policy combining algorithm was extended, to give priority to personalized user policies over application templates.
- When multiple policies are applicable, the policy combining algorithm has to compute the resulting allowed QoC. At the moment the maximum allowed QoC level from all applicable policies is provided in the evaluation result. This combining algorithm can be

extended. A useful extension would be to make this algorithm (user) policy controllable as well.

- An attribute finder module for each context type over which context-aware policies may be defined is added, so that the PDP can request values for these context types whenever they are needed to evaluate policies.

4.5.5 Proof of concept

To test the extensions made to the reference Sun XACML implementation, a proof of concept was implemented. This consists of the PDP itself as well as different GUIs to set policies, request access via a PDP, and a simulation context source to set different context variables.



Patient name: Patient-One

Patient location: Hospital

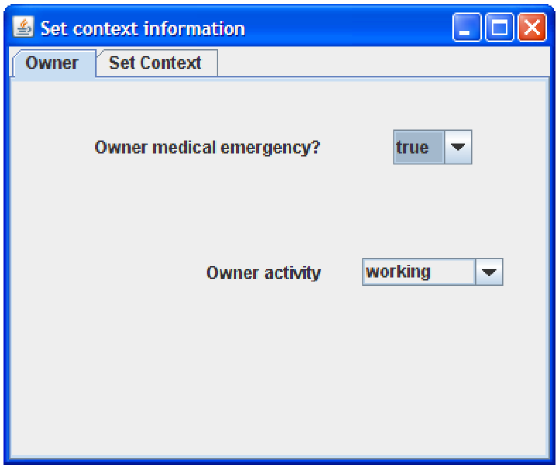
Patient activity: Dangerous

Caregivers in vicinity:

Patient in medical emergency: ☑ Yes/No

Figure 4-5: Adding a policy to the repository.

Figure 4-5 shows the interface for adding policies to the policy repository. Different attributes for the patient can be set, determining the situation (context) when this policy is valid. Also attributes of the caregiver (requester) can be set.

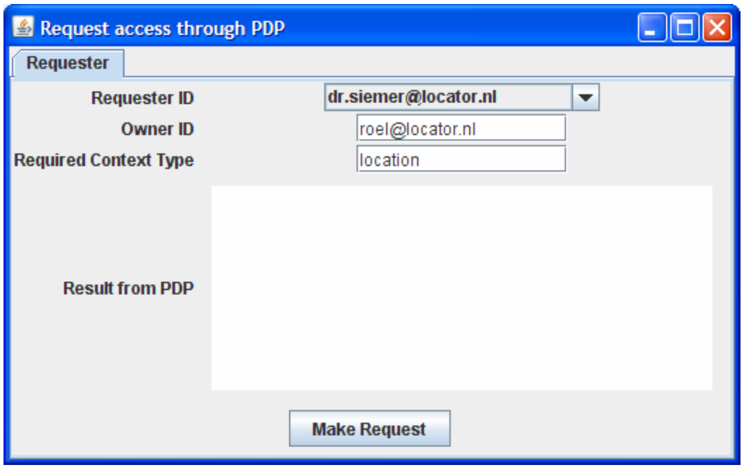


Owner medical emergency?: true

Owner activity: working

Figure 4-6: GUI for setting context variables.

The proper execution of the policies can be tested by first setting some context variables (Figure 4-6), after which the context itself can be requested via the PDP with another dialog box, shown in Figure 4-7.



Requester ID dr.siemer@locator.nl

Owner ID roel@locator.nl

Required Context Type location

Figure 4-7: Requesting context via a PDP.

Depending on the context variable and other information set via the previous dialog box, as well as the relevant policies, the results returned by the PDP (from the simulated context) will differ.

5 Summary

We started our research into perceived privacy in ambient intelligent environments in an exploratory fashion, followed by a targeted study that involved limited implication effects and continued by studying different application specific contexts. Within this approach, we limited the scope of the studies to the specific environment provided by the Amigo project to provide input and guidance to the design of the Amigo system. During the exploratory study, people were asked about their daily communications and related privacy issues. In the field study, a system for presence detection in the home and sharing that information across homes was developed and evaluated. The conceptual design study was conducted to find out how people would like to be able to mask or hide information that is being shared between different parties for three different types of applications. The main conclusions are:

- people use many diverse mechanisms to preserve their social privacy,
- people will share their personal information only with a small group of close relatives and friends,
- information sharing should have a clear benefit for users,
- users should have the possibility to control the level of detail of the information that is being shared, and
- users need a feeling of being in control, for example, automatic location detection is appreciated by users, but they also need to be able to influence the automatic detection mechanism.

5.1 System Design Implications

Although the qualitative and quantitative results and observations from the user studies provided a wealth of information on the perceived privacy of users, they didn’t provide information on how data should be secured, stored, or encrypted to support and advice application development. Initially, it was proposed to handle privacy at the middleware service level of the Amigo architecture by means of a rule-based filter that incorporated the user’s preferences and that would use the preferences to either pass on the data or not. Our field and concept studies showed, however, that such a mechanism for privacy filtering on the Amigo services level is not sufficient. Although there is definitely a need for a component that handles and stores the user’s privacy preferences, it is not sufficient for protecting the user’s perceived privacy, because it does not offer direct user control. Privacy should also be handled at application level. In particular, the type of information that is shared, the level of detail in which the information is shared, and with whom the information is shared (for example, with groups or individuals), are the most important concepts to take into account.

In addition to the implications for the system architecture, design guidelines could be derived from the results of the qualitative and quantitative studies to support the development of extended home environment applications. First of all, the most important rule to take into account is: ‘Maximize benefit, minimize effort and provide reasonable control for the end-user’. In addition to this rule, the following design guidelines need to be accounted for:

1. Provide proper security and inform users of security measures
2. Provide control on several levels
3. Present the user with a choice of level of detail in which the information should be shared
4. Provide clear feedback over shared information
5. Never automatically share information without user consent

6. Avoid using automatic intervention to maintain user privacy.

These guidelines were worked out with a detailed description, a general problem statement, examples from the Amigo extended home application scenario and a validation. Our guidelines are specific for applications in the extended home environment and focus on the sharing of data in a social context. This differs from existing guidelines for designing for privacy such as the OECD guidelines (Organisation for Economic Cooperation and Development, [27] and the guidelines from Langheinrich [20]. These guidelines are very generally applicable and refer mainly to the collection of data.

The guidelines were used to develop an application implementation. The goal was to build an application that shares different kinds of information between different users in different situations using (some of the) Amigo services. The environment and the privacy related settings that are needed from the user perspective were modeled. The user preferences for privacy control are dependent on the type of information that is being shared, the level of detail and with whom users are sharing this information. This means that user preferences are context-aware and privacy-aware. The model supports developers in specifying the user preference settings for different Amigo applications. We applied it in an Amigo home environment and evaluated the application implementation by means of an expert walkthrough. These assessments resulted in refinement of the original guidelines. This running prototype also provided a wealth of feedback with regard to privacy considerations. Since the system could demonstrate the consequences of changes in the context of people, their environment and activities, it could be used to explore the effects of being context-aware and privacy-aware.

The relation between system autonomy and privacy is another critical factor. Being able to cheat was one of the requirements revealed by the user studies. Setting different privacy preferences and changing them during a session elicited discussions on how this might affect people’s social relationships. With regard to the level of detail for the preference settings it was clear that different levels have different meanings for different people.

The different levels of precision that were maintained in the application for the disclosure of personal information needs further exploration. It is quite clear that people require control over their personal information and that they like to vary this level of detail depending on the context at hand. These levels are different for each individual. It appears that people require about 3 levels of detail, but that these levels are at different depth for each individual. These settings might potentially cause social embarrassments and conflicts.

In sum, the application implementation made it possible to explore and experiment with privacy settings in a context-aware environment. The results highlight the need for more exploration in different settings and conditions. But it also identified that some privacy controls should be handled in a generic fashion, such as employing privacy rules.

CAPriCE is a privacy policy framework that is well-suited to support application developers to author ‘policy templates’ which act as the default privacy configuration for their users. By using these templates, application specific defaults are provided. These templates are based on Quality of Context (QoC) concepts. These QoC indicators are: precision, freshness, temporal resolution, spatial resolution and probability of correctness. These indicators and their level of detail describe the privacy sensitivity of context information. By means of the Policy Decision Points (PDP) in the architecture, the QoCs can be mapped to the user-derived privacy model. Integration of these two models for other Amigo application should be further explored.

6 References

The references are organized per chapter.

Chapter 1

[1] Amigo Deliverable D1.2 (2005) Report on User Requirements Summary and Conclusions, Vol. I, Maddy Janse (ed.) et. al. IST-004182 Amigo.

Chapter 2

[1] Abrazhevich, D. (2004). Electronic Payment Systems: a User-Centered Perspective and Interaction Design, volume PhD thesis. Technische Universiteit Eindhoven. Altman, I. (1975). The environment and social behaviour. Monterey, CA: Brooks/Cole.

[2] Amigo Deliverable D1.1 (2005). Amigo User Research Results. Stefanie Un (ed.) et. al. IST-004182 Amigo.

[3] Amigo Deliverable D4.1 (2005). Report on Specification and Description of Interfaces and Services. Maddy Janse (ed.) et. al. IST-004182 Amigo.

[4] Baren, J. v., IJsselsteijn, W., Markopoulos, P., Romero, N., and de Ruyter, B. (2004). Measuring affective benefits and costs of awareness systems supporting intimate social networks. In Nijholt, A. & Nishida, T., editor, Proceedings of 3rd workshop on social intelligence design, CTIT Workshop Proceedings Series WP04-02, pages 13–19.

[5] Barkuus, L. and Dey, A. (2003). Location-based services for mobile telephony: a study of users privacy concerns. In Proceedings of the INTERACT 2003, 9TH IFIP TC13 International Conference on Human-Computer Interaction.

[6] Beckwith, R. (2003). Designing for ubiquity: the perception of privacy. Pervasive Computing, IEEE, 2(2):40–46.

[7] Bellotti, V. and Sellen, A. (1993). Design for Privacy in Ubiquitous Computing Environments. In Proceedings of the Third European Conference on Computer Supported Cooperative Work (ECSCW'93), pages 77–92. Kluwer.

[8] Burgoon, J. (1982). Privacy and communication. Communication Yearbook, 6:206–249.

[9] Cas, J. (2005). Privacy in pervasive computing environments - a contradiction in terms? Technology and Society Magazine, IEEE, 24(1):24–33.

[10] Chignell, M. H., Quan-Haase, A., and Gwizda, J. (2003). The privacy attitudes questionnaire (paq): initial development and validation. In PROCEEDINGS of the HUMAN FACTORS AND ERGONOMICS SOCIETY 47th ANNUAL MEETING.

[11] Consolvo, S., Smith, I. E., Matthews, T., LaMarca, A., Tabert, J., and Powledge, P. (2005). Location disclosure to social relations: why, when, & what people want to share. In CHI '05: Proceedings of the SIGCHI conference on Human factors in computing systems, pages 81–90, New York, NY, USA. ACM Press.

[12] Coyle, K. (2000). P3p: Pretty poor privacy? a social analysis of the platform for privacy preferences. Electronic Privacy Information Center and Junkbusters. http://www.epic.org/reports/prettypoorprivacy.html.

[13] Cranor, L. F., Arjula, M., and Guduru, P. (2002). Use of a p3p user agent by early adopters. In WPES '02: Proceedings of the 2002 ACM workshop on Privacy in the Electronic Society, pages 1–10, New York, NY, USA. ACM Press.

[14] Greef, P. and IJsselsteijn, W. (2001). Social presence in a home tele-application. CyberPsychology and Behaviour, 4:307–316.

[15] Greenfield, A. (2006). Everyware: The Dawning Age of Ubiquitous Computing. Peachpit Press.

[16] Hindus, D., Mainwaring, S. D., Leduc, N., Hagstrom, A. E., and Bayley, O. (2001). Casablanca: designing social communication devices for the home. In CHI '01: Proceedings of the SIGCHI

conference on Human factors in computing systems, pages 325–332, New York, NY, USA. ACM Press.

[17] Iachello, G., Smith, I., Consolvo, S., Chen, M., and Abowd, G. D. (2005). Developing privacy guidelines for social location disclosure applications and services. In SOUPS '05: Proceedings of the 2005 symposium on Usable privacy and security, pages 65–76, New York, NY, USA. ACM Press.

[18] Jabber Software Foundation (1999-2006). Jabber protocol. http://www.jabber.org/.

[19] Kumaraguru, P. and Cranor, L. (2005). Privacy indexes: A survey of westin's studies. ISRI Technical Report CMU-ISRI-05-138.

[20] Langheinrich, M. (2001). Privacy by design - principles of privacy-aware ubiquitous systems. In G.D. Abowd, B. Brumitt, S. S., editor, Ubicomp 2001: Ubiquitous Computing: Third International Conference, volume 2201, page 273. Springer Berlin / Heidelberg.

[21] Langheinrich, M. (2003). The dc-privacy troubadour - Assessing Privacy Implications of dcprojects. In DC Tales Conference.

[22] Lederer, S., Hong, I., Dey, K., and Landay, A. (2004). Personal privacy through understanding and action: five pitfalls for designers. Personal Ubiquitous Comput., 8(6):440–454.

[23] Margulis, S. T. (2003). On the status and contribution of westin's and altman's theories of privacy. Journal of Social Issues, 59(2):411–429.

[24] Meerling (1988). Methoden en technieken van psychologisch onderzoek, Deel 2: Data-analyse en psychometrie. Meppel: Boom.

[25] Neustaedter, C., Greenberg, S., and Boyle, M. (2006). Blur filtration fails to preserve privacy for home-based video conferencing. ACM Trans. Comput.-Hum. Interact., 13(1):1–36.

[26] Nunnally, J. (1970). Introduction to psychological measurement. New York: McGraw-Hill.

[27] Organisation for Economic Cooperation and Development (1980). OECD Guidelines on the Protection of Privacy and Transborder Flows of Personal Data.

[28] Price, B. A., Adam, K., and Nuseibeh, B. (2005). Keeping ubiquitous computing to yourself: A practical model for user control of privacy. International Journal of Human-Computer Studies, 63:228–253.

[29] Romero, N., van Baren, J., and de Ruyter, B. (2004). Design and assessment of an asynchronous awareness system. Technical Note 2003/00683, Philips Research.

[30] Rowan, J. and Mynatt, E. D. (2005). Digital family portrait field trial: Support for aging in place. In CHI '05: Proceedings of the SIGCHI conference on Human factors in computing systems, pages 521–530, New York, NY, USA. ACM Press.

[31] Schots, P. and de Regt, M. (2006). User-based Multiple-lab Evaluation. Number IST-004182.

[32] Sensite Solutions (2005a). Product Leaflet Logisphere BN208 Intelligent Tag. http://www.sensite-solutions.com/.

[33] Sensite Solutions (2005b). Product Leaflet Logisphere HBL100 Wireless Network Controller. http://www.sensite-solutions.com/.

[34] Simon, H. A. and Ericsson, K. A. (1993). Protocol Analysis : Verbal Reports As Data. Bradford Book, rev. ed. edition.

[35] Spiekermann, S. (2005). Perceived control: Scales for privacy in ubiquitous computing environments. In 10th International Conference on User Modeling.

[36] Tu, C.-H. (2002). The measurement of social presence in an online learning environment. International Journal on E-learning, 1(2):34–45.

[37] W3C (2006). Platform for Privacy Preferences (P3P) Project. http://www.w3.org/P3P/.

[38] Want, R., Hopper, A., Falco, V., and Gibbons, J. (1992). The active badge location system. ACM Trans. Inf. Syst., 10(1):91–102.

[39] Wikipedia (2006). Wikipedia. http://en.wikipedia.org/.

Chapter 3

[1] Oosterholt, R., Roberts, G., Putten, J. van der., Brouwer, A., Peeten, J., Neervoort, P., Kohar, H.(2006). EasyLogic 3.0 for TV centric products. The UI category Standard for Remote Controlled Large Screen Devices. Published by Philips Design.

[2] Sheikh, K., Wegdam, M., and van Sinderen, M.(2006). Enforcement of Dynamic Privacy Policies in Distributed Context-aware Homes. Adjunct Proceedings of the 4th International Conference on Pervasive Computing, Dubline, Ireland, Published by Australian Computer Society(OCG): Vienna, Vol. 207, pp. 227-230

[3] Lederer, S., Hong, J.I., Jiang, X., Dey, A.K., Landay, J.A., Mankoff, J(2003). Towards Privacy for Ubiquitous Computing, Technical Report UCB-CSD-03-1283, Computer Science Division, University of California, Berkeley.

[4] www.flickr.com

[5] Jabber Software Foundation (1996-2006). Jabber protocol. http://www.jabber.org

Chapter 4

[1] XACML core specification, http://www.oasis-open.org/committees/download.php/2406/oasis-xacml-1.0.pdf

[2] R. Hull, B. Kumar, D. Lieuwen, P. Patel-Schneider, A. Sahuguet, S. Varadarajan, and A. Vyas. Enabling Context-Aware and Privacy-Conscius User Data Sharing. In Proc. of the International Conference on Mobile Data Management, pages 187–198, IEEE, 2004.

[3] M Tentori, J Favela, VM González, Quality of Privacy (QoP) for the Design of Ubiquitous Healthcare Applications, Journal of Universal Computer Science, 2006 - jucs.org

[4] K.Sheikh, M.Wegdam, M.J. van Sinderen, Middleware Support for Quality of Context in Pervasive Context-Aware Systems, Proceedings of the Fourth IEEE International Workshop on Middleware Support for Pervasive Computing (PerWare'07), March 2007, New York, USA

[5] M. Wegdam, AWARENESS: a project on Context AWARE NEtworks and ServiceS, Proceedings of the 14th Mobile & Wireless Communications Summit 2005, 19-23 June 2005, Dresden, Germany.

[6] Halteren,. A., Bults, R., Widya, I., Jones, V., Konstantas, D., Mobihealth-Wireless Body Area Networks for Healthcare, Proc. New generation of wearable systems for e-health 2003, pp121-126, 2003

[7] T. Broens, A. van Halteren, M. van Sinderen, K. Wac, Towards an application framework for context-aware m-health applications, Proceedings of 11th Open European Summer School (EUNICE 2005), 6-8 July 2005, Colmenarejo, Spain

[8] EPAL: Enterprise Privacy Authorization Language, http://www.zurich.ibm.com/security/enterprise-privacy/epal/

[9] P3P: The Platform for Privacy Preferences, http://www.w3.org/P3P/

[10] E-P3P: Platform for Enterprise Privacy Practices, http://portal.acm.org/citation.cfm?id=644538

[11] Sun’s XACML implementation, http://sunxacml.sourceforge.net/

[12] Apache Xindice, native XML database, http://xml.apache.org/xindice/