Lappeenrannan teknillinen yliopisto
Lappeenranta University of Technology

Pekka Jäppinen

ME - MOBILE ELECTRONIC PERSONALITY

Thesis for the degree of Doctor of Science (Technology) to be presented with due permission for public examination and criticism in the Auditorium 1383 at Lappeenranta University of Technology, Lappeenranta, Finland on the 8th of June, 2004, at noon.

Acta Universitatis
Lappeenrantaensis
183

Supervisor Professor Jari Porras
Laboratory of Communications Engineering
Department of Information Technology
Lappeenranta University of Technology
Finland

Reviewers Professor Petri Pulli
Department of Information Processing Science
University of Oulu
Finland

Assistant Professor Dario Maggiorini
Dipartimento di Informatica e Comunicazione
Università di Milano
Italy

Opponents Professor Do Van Thanh
Department of Telematics
Norwegian University of Science and Technology
Norway

ISBN 951-764-902-9
ISSN 1456-4491

Lappeenrannan teknillinen yliopisto
Digipaino 2004

Preface

The work presented in this thesis has been carried out in the Laboratory of Communications Engineering in the Department of Information Technology of the Lappeenranta University of Technology, Finland, during the years 2001-2004.

I would like to express my gratitude to my supervisor Jari Porras for his support and interesting discussions during my work on the thesis. I gratefully acknowledge the contribution of Mika Yrjölä on prototype creation and co-authoring a publication. I would like to thank Kari Heikkinen for various fruitful conversations about personal information handling. I also like to thank the head of the Laboratory of Communication Engineering Arto Kaarna and my colleagues for providing creative and inspiring environment for doing research.

I wish to thank Professors Dario Maggiorini and Petri Pulli for reviewing the thesis and providing valuable comments.

Finally I would like to express my thanks to my family and friends for all the support and entertainment they have provided to me during these years. Without the balance they have provided to my life, this thesis would have never been finished.

Lappeenranta, April 2004

Pekka Jäppinen

Abstract

Pekka Jäppinen
ME - Mobile Electronic Personality
Lappeenranta, 2004
135 p.

Acta Universitatis Lappeenrantaensis 183
Diss. Lappeenranta University of Technology

ISBN 951-764-902-9
ISSN 1456-4491

Digital services require personal information for a variety of reasons. Due to advances in communication technology, new types of services are evolving along with traditional Internet services. Due to the diversity of services, the traditional approaches to personal information handling designed for Internet services are inadequate. Therefore, new approaches are necessary.

In this thesis, a solution where personal information is stored in and accessed from the user's mobile device is presented. This approach is called Mobile Electronic Personality (ME). The ME approach is compared to the existing approaches which rely on a database either at a service, a trusted third party or a client program. Various personal information properties are taken into account in the comparison of storage locations.

The thesis presents both the internal and the communication architecture of the ME. The internal architecture defines how the information is stored in the mobile device. The communication architecture defines how the information can be accessed by different types of services from the ME.

The use of the architecture is described for services in different environments. A simple ME based solution for the authentication of a user is defined. The authentication of service, which is required to protect the privacy of the users is also presented.

Keywords: personal information, identity management, Internet services, mobile services, mobility, personal trusted device, Bluetooth

UDC 004.5 : 004.738.5 : 004.056

3G Third generation mobile telephony system

AP Access Point

BDADDR Bluetooth Device Address, unique hardware address for Bluetooth device

CA Certificate Authority

CRAB Challenge-Response Authentication over Bluetooth

DOM Document Object Model, the way to describe documents as group of objects

ECML Electronic Commerce Modeling Language

FINEID FINnish Electronic IDentification

GOEP Generic Object Exchange Profile

GPRS General Packet Radio System

GSM Global System for Mobile Communications

HTML Hypertext Markup Language

HTTP Hypertext Transfer Protocol

IEEE Institute of Electrical and Electronics Engineers

IETF Internet Engineering Task Force

IP Internet Protocol

IrDA Infrared Data Association

ISP Internet Service Provider

MAC Media Access Control

ME Mobile E-Personality, Mobile Electronic Personality

MIME Multi-Purpose Internet Mail Extensions

NDS Novell Directory Service

OBEX Object Exchange

P3P Platform for Privacy Preferences

PDA Personal Digital Assistant

PIM Personal Information Management

PTD Personal Trusted Device

SAA Service Accessing Application

SAD Service Accessing Device

SDP Service Discovery Protocol

SIM Subscriber Identification Module

SOAP Simple Object Access Protocol

SSL Secure Sockets Layer

SSO Single Sign-On

SyncML Synchronization Markup Language

TTP Trusted Third Party

UMTS Universal Mobile Telecommunications System

URL Uniform Resource Locator, Universal Resource Locator

W3C World Wide Web Consortium

WAP Wireless Application Protocol

WLAN Wireless Local Area Network

WSP Wireless Session Protocol

XML Extensible Markup Language

XNS eXtensible Name Service

I Jäppinen, Pekka., Porras, Jari.
"Analyzing the attributes of Personalization information Affecting Storage Location", in *Proceedings of the IADIS International Conference e-society*, pp. 48-55, Lisbon, Portugal, June 3-6, 2003

II Jäppinen, Pekka., Porras, Jari.
"ME: Mobile E-Personality" *WSEAS Transactions on Computers, Volume 2, Issue 2,* pp.471-476 , April 2003

III Yrjölä, Mika., Jäppinen, Pekka., Porras Jari.
"Personal information transfer from mobile device to web page ", in *Proceedings of the IADIS International Conference WWW/Internet*, pp. 485-492, Algarve, Portugal, November 5-8, 2003

IV Jäppinen, Pekka., Porras Jari.
"Applying Challenge-Response Authentication over Bluetooth for web services", in *Proceedings of Softcom 2003, 11th International Conference on Software, Telecommunications & Computer Networks*, pp. 758-761, Split and Dubrovnik, Croatia, Venice and Ancona, Italy, October 7-10, 2003

V Jäppinen, Pekka., Porras Jari.
"Transfer of Personalisation Information from mobile Device to Transparent Services" in *Proceedings of IASTED International Conference on Computer Science and Technology*, pp. 321-324, Cancun, Mexico, May 19-21, 2003

VI Jäppinen, Pekka., Porras Jari.
"Flash Notes over Bluetooth Wireless Technology", in *Proceedings of the IEEE International Conference of Wireless LANs and Home networks* pp. 91-99, Singapore, December 5-7, 2001

In this thesis, these publications are referred as *Publication I*, *Publication II*, *Publication III*, *Publication IV*, *Publication V* and *Publication VI*.

CHAPTER I

Introduction

"*The day is coming when telegraph wires will be laid on to houses just like water or gas – and friends will converse with each other without leaving home.*"
- Alexander Graham Bell [1]

In 1876 Bell had his vision of bringing the communication services close to the customers to make them easier to use. Today, more than a hundred years later, the goal is still the same: to make the use of services easier for the customer. As many services require some information about the customer, this thesis concentrates on simplifying the method for providing personal information to the service from the customer's point of view.

Services need personal information to adapt the service to appeal to the customer. This adaptation is referred to as personalisation. Besides personalisation, services such as Internet shops, hotel registration, competitions, conferences, etc. require the user's personal information in registration to be able to provide their services properly.

The past ten years have changed the field of service providing rather radically. Prior to the late 1990's the telecommunications and data communication networks were separate. Customers used a limited set of devices dedicated to a given network type to access the services provided by the given network. It was not possible to access services provided in the Internet via mobile phone. Due to the convergence of telecommunication and data communication networks, the services could be accessed with a greater variety of devices. The prior concept of vertical access, where the accessing device, access network, communication protocol and service all rely on each other was changed into the horizontal model. In the horizontal model the communication device, access device and service are no longer tied tightly together. Instead, one customer can access the Internet service from a desktop computer using the Ethernet while another customer uses his mobile phone and the GSM (Global System for Mobile Communications) network. [2, 3, 4, 5]

The current situation, where we have a variety of access devices, access networks and services (see Figure 1.1), which can form different combinations for service accessing, also affects the handling and transfer of personal information. Some of the combinations benefit more from personal information transfer through personalisation whilst other

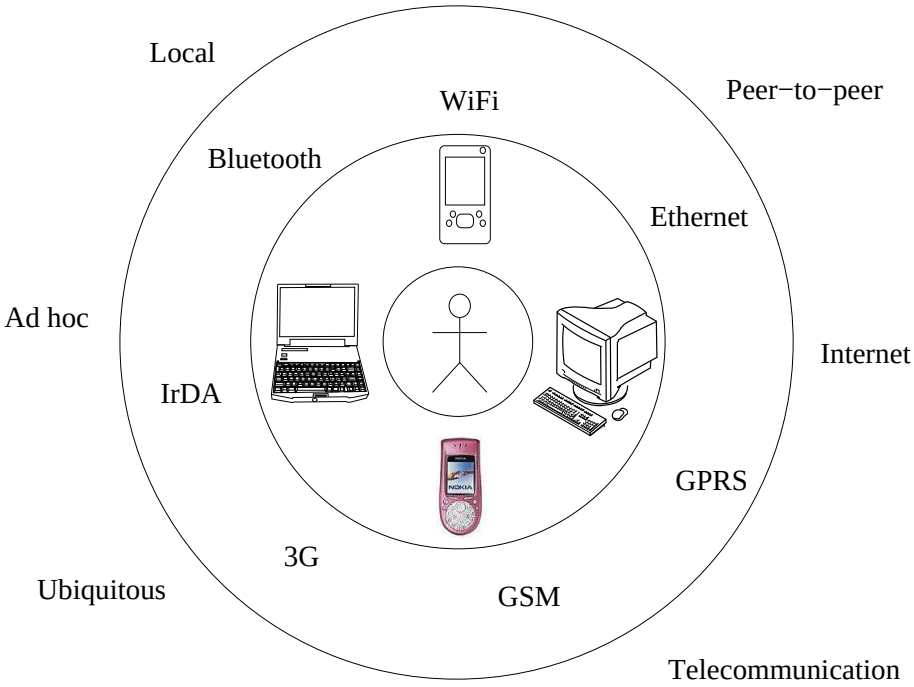

Figure 1.1: Multiple layers on services

combinations provide restrictions to the personal information transfer solutions. Providing a solution that would work in all cases will be challenging.

In the core of service accessing is the customer. The traditional customer uses Internet services from a desktop computer at home. This basic setting is quite stable: the access device is stable, the access location is stable and the access network is stable. When the customer changes to a laptop computer with wireless communication technology he can become nomadic [6, 7]. The customer is no longer tied to one access location but can move from one location to another and then form a new connection to the service. Customers anywhere in the world can take advantage of the variety of Internet access points which are provided by community networks or by commercial Internet Service Providers (ISP) [8, 9].

Some nomadic users do not carry an access device with them, but instead use publicly provided devices. For example, the customer may decide to walk into an Internet café and access the service from there. Therefore the accessing device is no longer stable and thus the solution for personal information handling cannot rely on the service accessing device.

Besides traditional computers, modern game consoles such as Playstation2 or Xbox can be used to access Internet services. As the primary design goal for consoles have been to provide a convenient interface for gaming, they have a very limited capability for inputting data (i.e. a keyboard is not included by default). A similar problem exists with digital television as with an Internet access device that is used with remote control.

They are useful for browsing through information, but filling in registration forms can be frustrating for the customer.

The limited input capability also affects most mobile phones and Personal Digital Assistants (PDA), which can also be used to access services. Besides limited input capability, these small mobile devices also have a small screen that cannot show as much information as a monitor or television. Therefore, methods for limiting and filtering the information the services provide are required. Effective customisation of information requires personal information and is called personalisation [10, 11, 12].

Besides the accessing device also the communication technology brings limitations. Mobile telecommunication technologies such as GSM and GPRS (General Packet Radio System) have a rather low bandwidth. Therefore, big images and large quantities of information are very slow to download. On the other hand, high latency has to be taken in to account in time dependent services i.e. in games. Furthermore, in most cases the customer also has to pay for the use of the communication medium based on the amount of data transferred, i.e the customer is charged for both the service and the data transfer [12, 13].

The limitations to the user interface and the communication medium increases the benefits gained from personalisation. On a desktop computer with a fixed connection it can be acceptable that the user is required to browse through several pages to get the information he is looking for. When the service is used with a mobile phone with a limited screen and data input mechanism and communication is carried out over GSM where every packet is charged, the required service should be available easily and with minimum communication [10, 12].

Besides the costly telecommunication technologies, the mobile devices provide a possibility for short range communication where a telecommunication operator is not needed. Infrared communications based on IrDA (Infrared Data Association) have been used for communication between the mobile device and the personal computer. Infrared has the problematic line of sight requirement and thus short range radio frequency based communication technologies have been developed. Most of the new mobile phones include Bluetooth capability as the short range communication option.

The free short range communication offers new possibilities for providing new types of services in addition to traditional Internet and telecommunications services. First of all, the short communication range makes it easier to provide location based services, as the user has to be in the perimeter of the given service access point in order to access the service [14].

Since communication technology does not rely on existing communication networks like the Internet, local independent services can be created. For example, a restaurant can provide within its premises a menu service, from which the customer can get the daily menu on his mobile device. The restaurant keeper does not need to buy an Internet connection, just a few access points connected to the server is enough. Since there is no connection to the Internet, the personal information handling cannot rely to the resources provided by the Internet.

Besides services provided in a fixed location, it is also possible that mobile device owners provide services to each other. The connection between terminals can be created by using

ad hoc networking [15, 16]. Thus we have gone through the information services that are accessed and provided from a fixed location all the way to mobile services accessed by mobile users.

But information services are not the only type of services that need personal information. Ubiquitous or pervasive computing provides a possibility for ambient environments [17, 18]. If user's personal information is gained, the environment can then be customised by the user likings. For example, a waiting room can choose the music played based on the taste in music of the people inside the room.

1.1 Scope of the thesis

In this thesis, a Mobile Electronic Personality (ME) is described. A ME is a service that resides in a user's mobile device, from where various types of services can request user's personal information using wireless communication. The goal for a ME is to allow the user more control over his personal information, i.e. better privacy as well as to provide better usability for the services by making the process of personal information delivery easier. The ME concept has many different aspects that have to be taken into account in order to generate a robust and complete solution. This thesis describes the main concept and concentrates on the question of how personal data can be transmitted safely from a mobile device to a service. The design emphasises the customer's point-of-view even though database optimisation, user interfaces and other important tasks for service use are left for further research and are not within the scope of this thesis.

The thesis is divided into seven chapters. In chapter 2, the personal information relations to the services is discussed and different approaches for personal information storage and transfer are evaluated. Chapter 3 describes the actual Mobile E-Personality architecture concentrating on personal information transfer from a mobile device to various services. Chapter 4 describes how the defined architecture can be used in different types of services. Chapters 5 and 6 conclude the thesis providing the results and discussion about further research.

1.2 Research methodology

The conducted research can be divided into three parts. The first part is the conceptual-analytical research where the current situation of personal information handling is evaluated. The evaluation points out the problems in current approaches of storing and transferring personal information. The second part is the constructive research, where new architecture is developed to solve the problems found. In the last part the developed architecture is evaluated with the help of case studies and prototyping.

CHAPTER II

Services and personal information

The term *personal information* is not unambiguous; it can mean several types of information. Some articles (for example: [19, 20]) that discuss personal information management consider personal information as *information that is owned by a given person*, such as calendar notes, contact addresses of the friends and so on. Similar interpretation is used also on mobile phone marketing. For example, Nokia's mobile phone features include Personal Information Management (PIM) which is clarified as calendar, contacts, notes, and to-do list [21]. In the context of services (and therefore in the context of this thesis), personal information means *information about a given person*, e.g. name, address and personal preferences and likings, such as a preference for science fiction books.

Services have two different needs for personal information:

1. To provide certain services. It would be impossible to deliver a book that a customer ordered without knowing anything about the customer.
2. To adapt the service to the user, i.e. personalising the service.

In the first case, the need for personal information is evident. No information means no service. E-shops require the customer's address to deliver the ordered items to the customer as well as billing information to get the monetary compensation for their service. Therefore, the user has to trust the service provider to handle the given personal information appropriately if he wants to get the service.

The second case is more complicated. The information is not needed to provide the service, but to adapt it to the user. The goal of personalisation is to benefit the user so that he uses the service more often and becomes a regular customer [11]. The more information the service gets the more it can adapt itself to the customer.

An important difference between the two cases is that in the first case, the user has to give identifiable information about himself. For the latter the identity of the person is not important, just what his preferences are.

2.1 Personal information handling

Digital services collect a large amount of information about their customers. There are several databases on users of digital services that contain a variety of information about them. With advances in data mining, pattern recognition, content based retrieval and in general the power of computers, there is a distinct possibility to get more accurate information about the customer and we move towards the World Without Secrets [22]. Therefore, the personal information should be handled so that the privacy of the customer is not jeopardised.

The initial source of this information is the customer. The information can be acquired explicitly, i.e. asking the user directly, or implicitly, i.e. indirectly by following the user behaviour in the service [23, 24, 25]. To accomplish this, current services require users to register themselves into the service. The acquired information is then stored in user-models on the service database. From there the service accesses the information in order to personalise the service.

Initially the user is anonymous to the service, but when personal information is transferred to the service provider the nymity of user changes. Goldberg defined in his PhD thesis four types of nymities: verinymity, pseudonymity, linkable anonymity and unlinkable anonymity [26]. For a service provided in the Internet, the customer has unlinkable anonymity before any information is exchanged. With a cookie, the service can link the customer to his previous actions on the service, thus changing the user's nymity to linkable anonymity. If the user now creates an account to the service with funny username, the user becomes pseudonymous towards the service. Finally, if user gives detailed personal information, he is verinymous, i.e. the user's identity can be identified and verified.

According to LaRose and Rifon, the most common personal information requested by the service is the user's email address and right after comes the name of the user [27]. This means that in most services the user has to give out his identity. On the other hand, knowledge of the user's identity is not necessary to provide personalised service to the user. In their research Kobsa and Schreck defined methods to improve user privacy through pseudonymity [28, 29].

Due customers' concern about their privacy, privacy seals have been introduced as a method for recognising a trustworthy service provider. There are several seal providers and they have their own criteria for granting the seal. In their research, LaRose and Rifon notified that the services holding a seal were more likely to request more personal information than those without a seal. However, the policy statements of the services differed very little [27].

Services on the Internet are provided globally. The concept of privacy differs from culture to culture [30]. This means that the ways the personal information is handled differ. For example, in the United States companies own the data they collect about their customers and can sell it to other companies. In the European Union, the organisations that collect data about customers have to register themselves with the government. They also have to tell the customers why the information is collected and they are responsible for keeping the information secret [31].

To minimise the risks related to customer privacy, the services should require only the information they actually need. If the information is stored in a service database, the database should be encrypted and the users of the database authenticated. Furthermore when the data is transmitted the transmission media should be protected.

2.2 Personal Information Storage

Digital services can be provided in various environments to various types of devices in various ways. With such a diversity of equipment and environments, a generic architecture for the service use has to be defined.

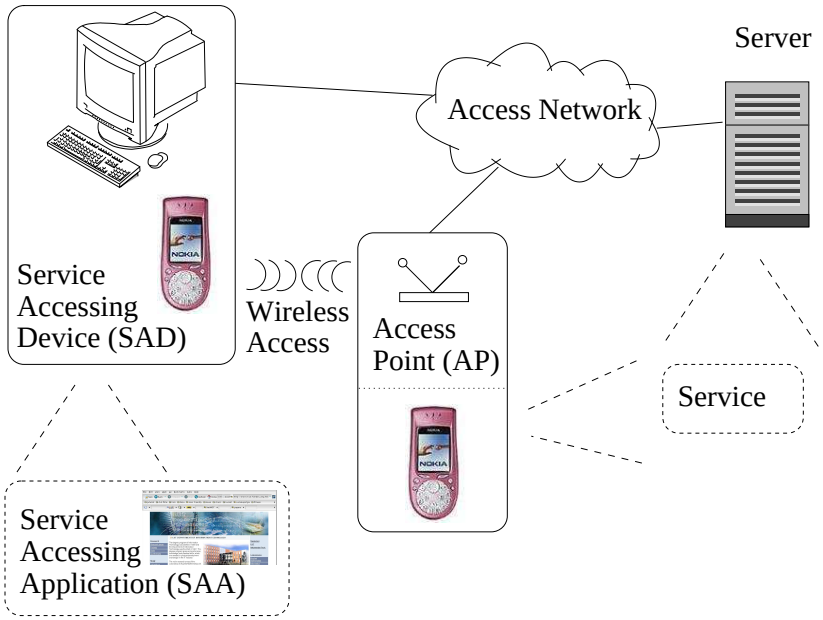

Figure 2.1: General service architecture

Figure 2.1 shows the general architecture for service access which has both wired and wireless access to the services.

The hardware parts of the architecture are:

- *Service Accessing Device* (SAD): The device that contains the application that is used to access the service for example a web browser. This device can be either mobile (mobile phone, PDA, laptop) or fixed (desktop computer).
- *Access Point* (AP): Mobile SAD connects to the access point in order to get access to the access network and the service. Sometimes the service itself can be in the access point, for example when a mobile phone provides a service to another mobile phone (ad hoc service).

- *Access Network:* Usually a public network that is used to transmit data from an SAD to a service.
- *Server:* The place where the actual service resides. A server usually has a great amount of computational power so that it can serve multiple clients.

The user accesses and uses the service with a *Service Accessing Application* (SAA) that resides in the Service Accessing Device. The SAD connects to the server where the *service* resides through an access network. When using wireless communications the connection to the Access network is done through an Access Point. In some cases the AP itself can provide the service.

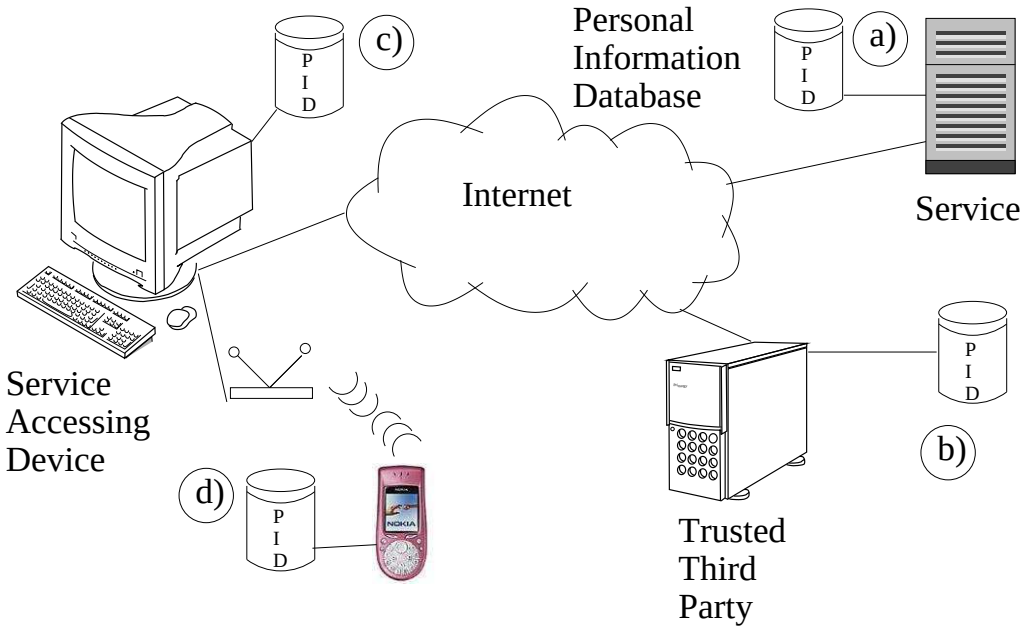

Figure 2.2: Personal information storage possibilities

Thus the three parts capable for storing personal information are the user, the network and the service. In the service, the service provider has created its own database about its customers (Figure 2.2, a). In the network, the information is stored in an external server from where the service can request the information. This external server can be a customer controllable website or a dedicated service for personal information storage provided by a Trusted Third Party (TTP) (Figure 2.2, b). On the user side, the information can be either in the service accessing device (Figure 2.2, c) or an external device from where the SAD can access the information (Figure 2.2, d).

2.2.1 Storage at the service

Currently, the storage place for personal information is usually at the service end. This approach is easy to implement for the service provider since it does not require any

special software for the user or special arrangements with other parties. The first time user has to register at the service and give out the personal information requested. The information is then stored in the service database where the service can access it when needed.

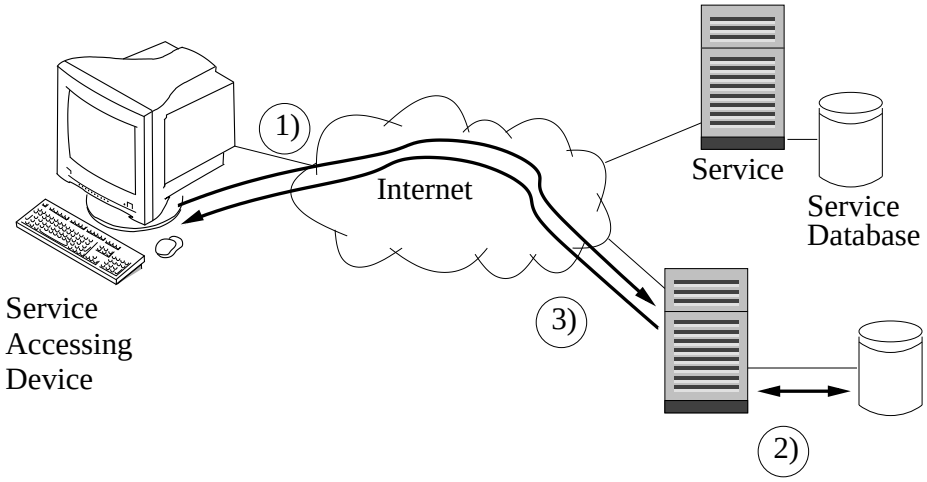

Figure 2.3: Personal information stored at service

The general service use scenario with personal information stored at the service is described in Figure 2.3 and consists of three steps:

1. The user forms a connection to the service and identifies himself.
2. The service checks if there is information stored about this user in its database
3. The service provides personalised service to the user.

This approach has some significant drawbacks. Since the services are more or less independent from each other, the user has to type in the required information every time he decides to use a new service. For example, when taking a trip to three different towns the user has to reserve a hotel room for each town. This means that if the hotels have no co-operation, the user has to type in his personal information three times in a very short time.

Without service co-operation the same personal information ends up to be stored in several databases. The more services the user uses, the more databases hold his personal information. The more databases that hold the personal information, the bigger the risk that it leaks from one of them. Although in general the service providers are trustworthy, they might not posses the skills to protect the information from a skilled hacker. Also, databases holding personal information about several persons tend to attract criminal minds more than databases holding just information about few persons. As Bruce Schneier stated in the cryptogram webzine: "*The real risk to personal data are the large databases at the end points, not the communication between them*" [32].

In time some of the personal information tends to change. Since the information is stored in several databases, the change in data has to be updated in all of them. In practise this means that there will be several inaccurate databases and profiles about one user.

In order to use the personal information, thw service has to recognize the user, i.e. the user has to authenticate himself to the service. Therefore, the user can be at best pseudonymous towards the service. The authentication process usually requires extra effort from the customer, e.g. the user has to type in his pseudonym and corresponding password. If the customer just wants to browse through the inventory of the webstore, the extra effort and loss of anonymity are not desired by the customer.

Besides the problems from the customer's point of view, there also exists problems from the service provider's point of view. Storing personal information about various customers requires a lot of storage space and an effective database system. Since the service holds a database about persons, the company hosting the service has to know the privacy laws of the given country [33]. Multinational companies may have to do extra work or even create extra customer databases for its services that reside in countries in which privacy laws differ radically.

2.2.2 Storage at the network

The wireless research forum's book of visions presents an idea that personal information should be uniformed in such a way that many applications can use the same information and thus there is no need to store it many times [34]. This information was recommended to be stored on one central server on the network.

The network based approaches can differ somewhat from each other. In principle there are two different possibilities. Either use a trusted third party who holds the information or have service provided on the user's own network, e.g. a web page.

This centralised server can be provided by a trusted third party. The use of the third party approach can be divided into 5 steps shown in Figure 2.4.

1. The user connects to the service and identifies himself
2. The service requests the user's information from the trusted third party
3. The third party checks the user's identity and fetches the requested personal information from the database
4. The third party provides the corresponding data for the service
5. Service provides the personalised service

Integrated Personal Mobility Architecture [35] defines a framework where the user's personal information is requested from a "home" network by the network which the user is visiting. This approach relies on the fact that the service can connect to the user's home network. This may not be possible for services provided ubiquitously.

Skender and Saric propose that the user profile containing the information about the user be stored in a central database "where it can be monitored by user and accessed

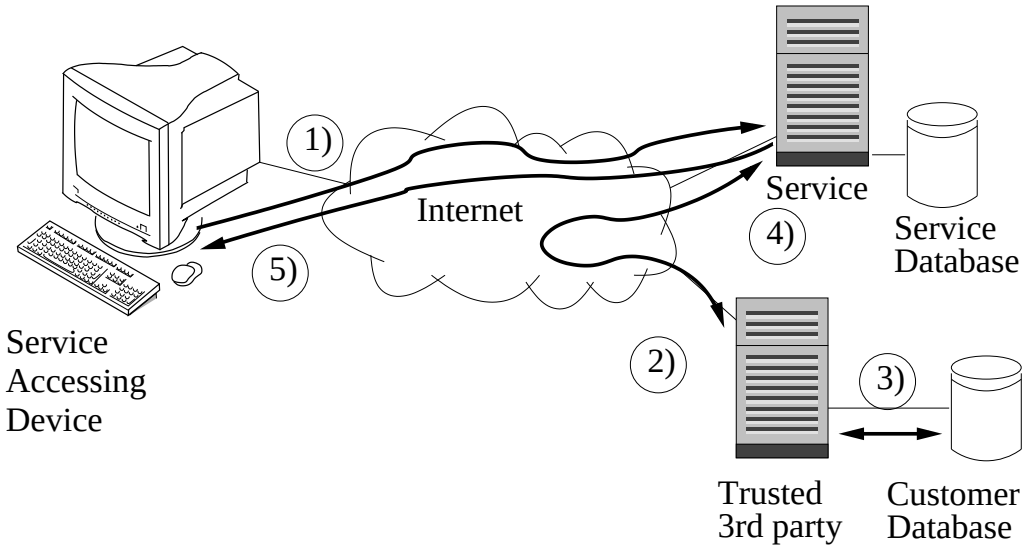

Figure 2.4: Personal information stored at third party

by different services" [36]. The approach is designed for GSM networks on which the operator stores the user profiles.

Koch and Wörndl defined the ID-Repository, which contains the user profile and can be maintained by the user [37]. The repository is located on the server of the identity provider. The ID-repository approach also allows service providers to add data into the user profile.

For a TTP approach there exists single sign-on (SSO) architectures that can be used also to transmit some personal information to the service. Three different approaches to this are from big players in the industry: Microsoft .NET Passport, Novell's DigitalME and Liberty alliance's liberty architecture.

In the .NET Passport approach, the user registers at the Microsoft server giving at least an email address and a password for the passport. When accessing .NET Passport enabling services, the user first authenticates himself to the Passport server. The passport server then handles the authentication of the user for the services [38]. To use .NET Passport as an authentication method for services the company currently has to pay $10,000 for a one year license [39]. For big companies this is not a problem, but for companies providing services with little value, the price tag is quite high. It is also unlikely that conference organisers are willing to do the extra work required to join the single sign-on system especially when all they need is simple registration information for a one-time event.

Novell DigitalMe is similar to the passport but is designed to work with NDS (Novell Directory Service). It also allows the customer to create several different profiles called meCards, which describe the person. The customer can decide which meCard is delivered to the given service or person. DigitalMe can be integrated into the Internet Explorer toolbar from where it can be used to fill in forms on Web pages. DigitalMe identities can

be also used in AOL instant messaging for sharing contact information with a communicating partner. DigitalMe is free for the customer, but it currently supports only the Windows environment [40].

In Liberty architecture there are identity and service providers that form circles of trust. User identity and information is transferred between the participants of the circle when necessary. Compared to the .NET Passport, the identity is controlled by a multitude of companies instead of just one [41]. Forming of circle of trust requires negotiations between the participating companies. Therefore, it is unlikely that there will be one circle of trust that contains various sizes of companies [42]. As in .NET Passport, this is not suitable for one time services provided by small service providers.

In order to request the personal information the service requires a connection to the information storage location, e.g. TTP. In practise this means that ad hoc services provided by other mobile devices or local transparent services that do not have a connection to Internet cannot use the network based approach. This obviously diminishes the usability of the approach.

2.2.3 Storage at the user end

The final possibility is to store the information at the user end, where the user has all the control over the information. On the user side, the information can be stored either at the service accessing device or at the external device from where the information is then requested. Both approaches require support from the application that is used for accessing the service, e.g. a Web browser.

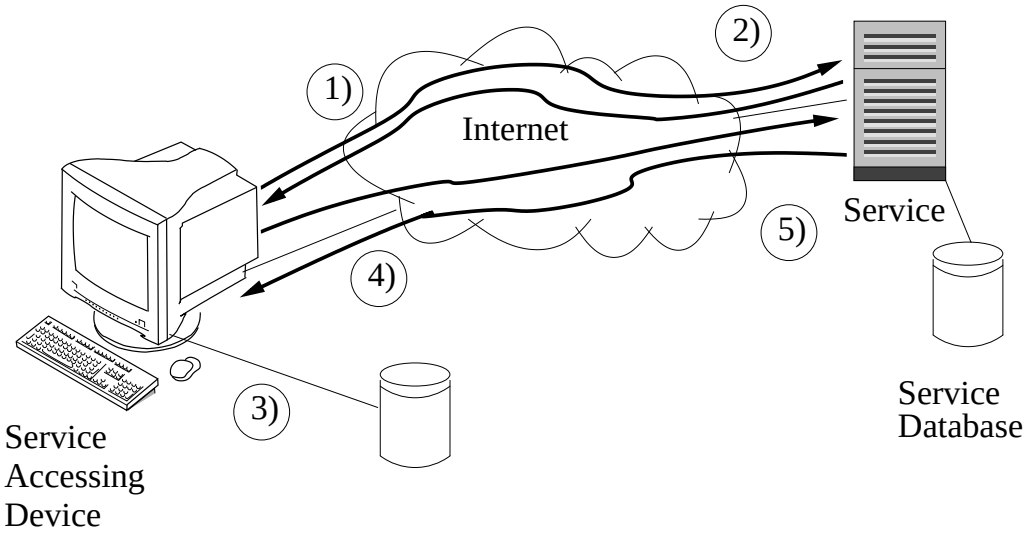

Figure 2.5: Personal information stored in web browser

The storage in the SAD approach is shown in Figure 2.5 and consists of five steps:

1. The User connects to the service.

2. The service requests the personal information.

3. The service accessing application, e.g. the Web browser, gets the information from its database .

4. The software provides the information.

5. The service provides the service

For example, the Mozilla wallet can fill Web forms automatically when the form fields are notated properly [43]. The user information is stored on the database of Mozilla. This approach is fine as long as the user uses only a personal computer at work or at home. For a mobile user that uses mainly either Web cafés to access Internet services or ubiquitously provided services, which are not accessed by Web browser at all, Mozilla wallet helps very little.

External device use provides the user mobility. Intel researchers have defined a Personal Server [44] that is a device where the user can store different kinds of information. The device itself has no user interface and the information has to be accessed wirelessly through other devices. Personal server information can be accessed using various web protocols such as HTTP (Hypertext Transfer Protocol) and SOAP (Simple Object Access Protocol). Since the Personal Server is designed mainly to be a plain information storage facility for the user it does not for example define how information can be automatically accessed from it by a trusted service.

While Liberty architecture provided company federated identity, XNS (eXtensible Name Service) aims to do the same with community governed identity [45].

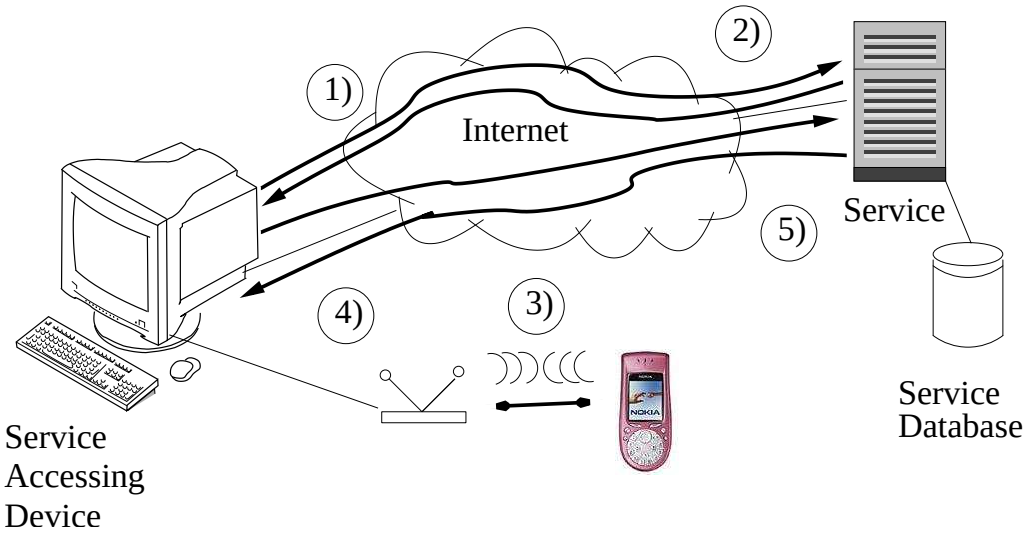

Figure 2.6: Personal information stored at the external device

The storage in an external device scenario shown in Figure 2.6 consists of five steps:

1. The user connects to the service.
2. The service requests the personal information.
3. The service accessing application, e.g. Web browser, forms a connection to the external device and requests the information.
4. The software provides the information.
5. The service provides the service.

The goal of the ME is to define the ways of communicating with the mobile device holding personal information. The access to information in the ME is designed so that after configuration the user's actions are minimised but the privacy is preserved.

Mobility based problems can be solved when the user carries a device containing the personal information in itself. The services can then request the required information from this single device. When using trusted SAD, the external device and SAD can synchronise their information to provide better response times.

2.2.4 Comparison of storage locations

As the storage locations differ from each other, so do the pieces of personal information. The various properties of personal information were evaluated in *Publication I.* Below are described the most important properties from the storage location point of view.

Before any information is requested it is important to evaluate whether the information is actually *needed* or not. Often the personal information is requested "just in case", i.e. the service provider thinks this information might be needed for something. Surprisingly, Web sites that have a seal that proves they honour customers' privacy are more likely to request information that seems to be unnecessary for them [27]. Customers who feel that a service requests unnecessary information may provide false information just to be able to access the service. Some customers might decide to not use the given service at all. Better results can be achieved by carefully considering the actual needs and the properties of the required information [11].

From all the information that is needed to provide functional and useful services, there are many items that are needed only at the time when customer uses the service. Therefore, it is important to consider the *necessity for* information *storage.* For example, the information that a customer likes science fiction books is required only when customer browses the bookstore catalogue unless the store wants to send personalised advertisements to the customer. Thus the bookstore doesn't have to store the customer's preference in its own database, if it can be requested from somewhere else when needed. A responsible service provider also considers how long the information is needed. For example a customer's name and address may be needed in order to mail the ordered book to the correct customer. After the package has been delivered and charged those pieces of information are no longer important.

Another important issue is the *stability* of the information. Some information such as a birthday never changes and is thus stable. People move quite rarely and thus the mail

address can be considered rather stable information. The value of some information depends on other aspects such as time, location, role and situation. For example, when the user is on a business trip in Lappeenranta looking for a place to have lunch his preferred beverage at lunch is likely to differ from the one he has when he is on vacation with his wife in Barcelona. Thus the preferred beverage is not stable. There is also information such as bank account balance that will change frequently and is thus unstable. As to unstable information, the user has to be able to somehow easily update it.

In case of dependable information, the user has to be able to define which dependencies apply. For example Nokia 6130 has the possibility to define the phone to function differently for a set of defined situations, such as a meeting [46]. For Symbian OS based mobile devices there exists SmartProfiles program, which can be used to define various profiles that define the device behaviour when certain dependencies apply. For example the phone will not ring on weekends if the call is coming from a work-related phone number [47].

The different storage locations have different amounts of storage space, thus the *size of the information* is important. In the approaches which require a lot of communication a big amount of data slows down the service use. Currently the information needed for personalisation is based on textual representation and thus does not take too much space. Therefore, the size of the piece of information alone does not affect the storage location. In the future however, it is possible that new ways of describing personal preferences may arise. For example, a dating service might base its decision on a picture of the preferred looks of a dating companion instead of plain text information stating height, hair colour and so on.

Some information is so general that any service can use it, while other information is specific to the given service, thus the *generality of the information* has to be considered. General information such as name or address should be easy to provide for the services that require them.

Service-specific information originates from the service. But there can be generic information that the service generates. A restaurant can learn that its customer likes to eat fish on rainy days, while on sunny days he likes salad. While the *origin of information* is the given service, the information itself can be thought generic, as other services could benefit from it. The question is, can the created information be considered as a company secret, even though it is definitely customer information [11].

The level of privacy greatly differs according to the piece of information and the user. Users may have preferences concerning the required privacy level. Some information is very private for some users whereas other users may consider the same information public. Thus the final decision concerning the location of the information storage has to lie with the customer.

The different storage locations have their advantages and disadvantages which the aforementioned properties affect. Different storage location properties are compared in Table 2.1.

The storage capacity affects to the amount of personal data that can be stored at a given location. A mobile device has the smallest amount of space for storage. However, the amount is enough for text-based information. Furthermore, the storage capacity

Table 2.1: Comparison of storage locations

Location	Service	Network	Network/User/ SAD	User/ME
Storage capacity	GigaBytes	Gigabytes	100's of MegaBytes	10's of Megabytes
Whose data	The service customers data	The data of customers of several services	Single customer	Single customer
Amount of storage places	Many	Few	One	One
Cost to user	Free	Possible TTP cost	Possible server upkeep	Free
Cost to service provider	Database upkeep	Requires a deal with the TTP	Free	Free
Nymity	Pseudonymity	Pseudonymity (Anonymity)	(Linkable) anonymity	(Linkable) anonymity
Cold start	not possible	possible	possible	possible

of mobile devices will gradually increase and external memory cards can be used. For example the IBM microdrive, provided as compact flash card, can already hold more than a Gigabyte of information. When we also consider that the mobile device has to store data of only one customer while a service or TTP has to store data about many customers, it becomes evident that the storage capacity can be thought to be rather similar per customer in all approaches.

The amount of storage location directly affects the security risk related to personal information. The more the databases hold information, the bigger the risk is that at least one of them has a security flaw. Also, the more customers and their information the database holds, the more that given database attracts malicious people [48]. Therefore, the third parties are likely to be the main target of personality thieves. The security is not the only concern for the customer. When the customer's data is stored in several databases, its upkeeping becomes more cumbersome and thus in time it is likely that some of the databases hold outdated information.

When personal information is stored in one location, it is important that there is a common notation that can be used when requesting and providing information. With a common notation the services understand the meaning of the given data. Currently there is no single common standard for defining the various types of personal information.

Users are not likely to pay a lot of money just for their own personal data storage. Therefore, the user has to get something more from the third party if the third party wants to charge the user for its services. Big service providers might be willing to pay for information about their customers. Smaller ones on the other hand might not be able to afford the prices that for example Microsoft wants to charge for its services. This is especially true if the customer information is used only to provide better usability for a

low cost service. The extra cost for the service provider finally affects the price of the service given to the customer.

The actual user identity is not always necessary in order to provide personalised service [48]. For example, when a customer browses the pages of a bookstore, it is not important who is browsing but what kind of books he likes. When the information is stored in the service the user has to at least give the pseudonym he has used in registration in order to get personalised service. In the TTP approach the user has to be identified similarly by the service so that the service can request the correct information from the TTP. In theory, the identity could be a somewhat random ticket which the customer gets from TTP. Thus the user's identity is not revealed to the service. When the information is stored in SAD, the service can request the required information simply from SAA. In the SAD approach, the customer is tied to the given SAD and thus his connections can be linked from one connection to another by the IP (Internet Protocol) address or the use of cookies. A respectable service provider does not link the customer to the previous visits in the given service without the user's consent and thus allows full anonymity. Similarly in the external device based approach the ubiquitous service can use a device identity, e.g. a MAC (Media Access Control) address, to link the customer to previous service uses. When using a public SAD to access the service, the external device identity is not revealed and the user can be completely anonymous. Finally, the user's anonymity depends on the amount and type of data that is transferred to the service and whether the service has stored this data (with or without user consent) in its databases.

When the user connects to a service for the first time, the service does not have any knowledge about the customer. The customer still expects personalised service right away, which causes a so called "cold-start-problem". Without any information about the customer it is impossible to provide personalised service [25]. When the customer data is stored externally and the server can request it, the cold-start-problem can be resolved. The problem is especially harmful for customers using a mobile device with limited input capability as the personal information cannot be easily provided to the service.

2.3 Discussion

While this thesis concentrates on the approach where information is stored in a mobile device, it is not the optimal solution for all data. The information created and owned by the service should rather be stored at the service side. Also, the information needed by the service while the customer is not connected has to be stored at the service side. Several storage locations create the problem of data inconsistency. This can be solved by synchronising the databases when the customer connects to the service. The databases may also contain conflicting information. In general, the customer side database can be considered to be more accurate than a third party or a server side database. The newer data can be also identified by adding a timestamp to the data.

CHAPTER III

Mobile E-personality

In chapter 2, the storage of personal information and the advantages and disadvantages of different locations were discussed. From the various possibilities, the Mobile E-personality (ME) concept relies on the user's mobile device as the personal information storage location. The ME-service is the application running in the Personal Trusted Device (PTD). It handles the personal information and decides whether the information is delivered to the requesting service or not. ME communication architecture defines how services communicate with the ME-service.

PTD is a device that is assumed to be carried by the user all the time. PTD has to have the capability to store information and to communicate wirelessly with other devices. In this thesis, PTD is assumed to be a modern mobile phone which fulfills all the requirements and which people already carry with them most of the time.

ME can be used by different types of services to get information about the user. The goal for ME is to make it possible to provide personalised services in a variety of environments and still let the customer control the personal information flow, i.e. increase both the usability of service and the privacy of the user.

3.1 Services and ME

In ME, the personal information is stored in the mobile device. To understand the advantages and obstacles to creating such a service, the general architecture and different components needed for service usage have to be evaluated. Figure 2.1 defines the general architecture with hardware, software and communication components.

When defining the ME and its use it is important to realise the different ways that digital services are accessed. Figure 3.1 shows different types of services provided by a variety of devices over a variety of communication technologies.

From the ME point of view, these services can be divided into two main groups based on the access method: *direct* and *indirect* services. In indirect services, the service is

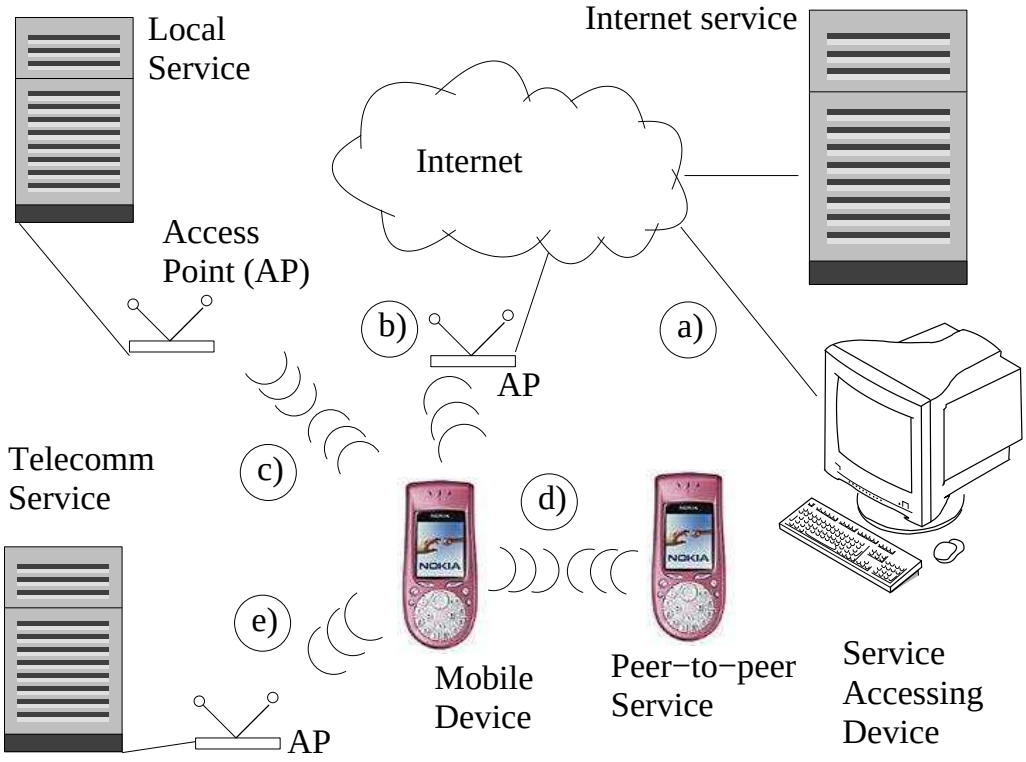

Figure 3.1: Service access methods

accessed by an external service accessing device, such as a desktop computer (Figure 3.1, case a). In direct services, the service and the mobile device that runs the ME-service communicate directly with each other (Figure 3.1 cases b,c,d,e). Direct services can be provided in a variety of ways. A customer may try to acquire the same Internet services with a mobile device as with the service accessing device (Figure 3.1 case b). The service can be a local service that is provided only locally. It can be a menu for a nearby restaurant or an interactive relaxation room that changes its wall colour based on the customer's preferences. Mobile devices can provide peer-to-peer services for each other (Figure 3.1 case d). One device owner may requests some information from another device owner. This information can be, for example, a list of ringing tones or the capability for starting a certain type of game. Telecommunication services (Figure 3.1 case e) are provided by commercial service providers. Besides the fee charged by the service provider, the customer also has to pay the telecommunications operator for the transmitted data.

Direct services can be further on divided into two groups: *transparent services* and *interactive services*. In transparent services the initiative to form a connection is done by the service. The service itself requires no action from the user either. One such service could be a guidance system [49] that utilises Flash notes, defined in *Publication VI*. The services push guidance notes to the customer's mobile device so that the customer knows which way to go. Interactive services require actions from the user and the initiative for service use comes from the user. For example, in services provided on the Web, the

customer creates a connection to the Web and requests a service, which is then provided for the customer.

3.2 ME communication architecture

The different access methods have to be taken into account when designing the communication architecture for the ME. How the personal information is requested depends on the way the service itself is accessed. How the three different service access types work on me is shown in Figure 3.2.

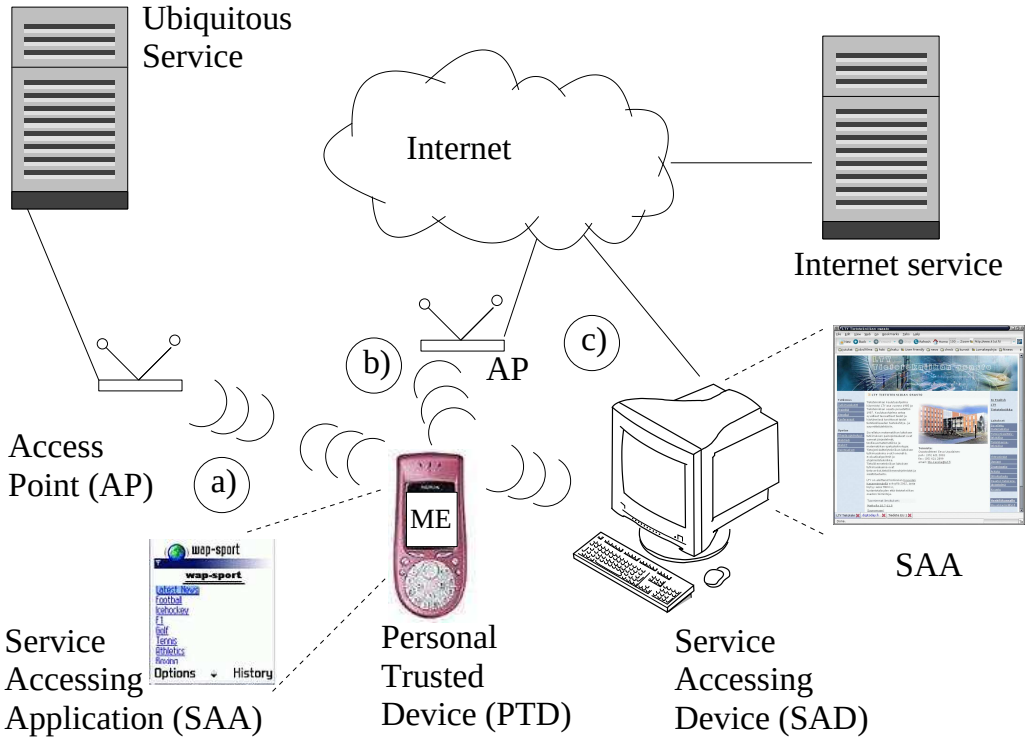

Figure 3.2: ME and services

1. The service is accessed with a program on the PTD. Usually the service accessing application (SAA) is some kind of web browser that uses either HTTP or WAP (Wireless Application Protocol) for communication with the given service (Figure 3.2, cases a and b). Thus the ME-service is accessed by the browser when personal information is required.

2. The services are accessed by SAA on an external service accessing device (SAD), such as a desktop PC, which can be for example at the library (Figure 3.2, case c). The SAD requests the required personal information from the PTD when necessary.

3. The services access the ME directly when they want to provide personalised service, such as a specialised advertisement or a note about a local event that is in the user's interest area (Figure 3.2, case a).

The different access methods affect to the way the personal information is requested from the ME. For interactive services, the personal information request is done by the SAA. The functionality can be implemented in directly on the program or be added as an additional component, i.e. a plug-in. When service is accessed with the PTD, the information can be requested locally while the use of an SAD means that the SAA and the ME need to communicate using wireless communication protocol. For transparent services the request for personal information is done by the service itself through a service access point in a similar manner than the SAD does.

3.3 ME and personal information

In the ME-concept, the personal information is handled in various ways. First of all, the information is *stored* in the mobile device. From there the personal information can be *requested* by different applications so that the required information can be *transferred* to a service.

In order to understand what a given piece of personal data means, some kind of notation is required. The notation is used for storing the information in the mobile device. In an ideal situation, the names for personal information data fields are used to request the personal information and then to notate the information when transferred to the service. As there are already many different notations it is sometimes necessary to translate the notation used in the ME into a notation supported by the service.

For basic contact information exchange, the Internet Engineering Task Force (IETF) has defined vCard [50, 51]. Most of today's mobile phones support vCard transfer over infrared link. SyncML (Synchronization Markup Language) [52] was defined to keep information, such as calendar notes, synchronized between various devices. Among other things, SyncML also supports vCard. Unfortunately there is no support in SyncML for all the personal information that is required for the ME [53]. Many of various digital wallets, whether in a Web browser or a mobile phone, rely on ECML (Electronic Commerce Modeling Language) when transferring credit card and delivery address information to the service provider [54]. The variety of different markups can be understood when looking at the amount of markup languages for name and address information that exist on Oasis consortium's Cover Pages notation library [55]. In addition many services have their own notation which they use to store customer data. These notations are adequate for what they have been designed for but are too limited for ME purposes.

One of the goals for the ME is to provide the user more control over the security of his personal information. Thus the security aspects of the personal information have to be considered before defining the notation. Many of the security measures will slow down the use of the service. Some of the information is non identifiable and so vague it alone causes no risk for the customer's privacy. Since it is impossible to know what information the given customer thinks is private and which is common, the user has to be able to define the level of secrecy required for each piece of information, i.e. the user

has to be able to manage his identity. This type of identity management is something we do in normal conversation everyday when we decide on what to tell one another about ourselves [56].

The previously mentioned aspects have to be taken into account when deciding how to notate the personal information. For the sake of simplicity, the same notation for pieces of information should be used when the information is stored in a database as well as when it is requested from the ME and transferred to the service.

Algorithm 1 Part of XML-schema for personal information

```
<xs:simpleType name="MESecLev">
    <xs:restriction base="xs:integer">
        <xs:minInclusive value="0"/>
        <xs:maxInclusive value="9"/>
    </xs:restriction>
</xs:simpleType>
<xs:element name="FullName">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="xs:string">
                <xs:attribute name="SecurityLevel" type="MESecLev"/>
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>
```

The notation is defined with the XML (Extensible Markup Language) schema [57] and part of the schema is shown as Schema 1. The schema has been kept quite simple. The tag names in schema identifying basic information follow the tag names used in ECML. Along the tag names there is an added attribute *MeSecLevel*. This parameter is used to define the security level of the given piece of information at the storage location. Its use is discussed more closely in subchapter 3.4.1 where the security agent and its functionality is introduced.

3.4 Internal structure of ME

When designing the internal structure of the ME-service various aspects have to be taken into account. The main goals for the ME are to increase the usability of personal information and at the same time increase information security by granting the user greater control over his personal information. Therefore, the information about the user stored in the ME has to be stored securely and should be easy to update. Due to the security reasons, the user of the PTD has to be authenticated before information in the ME-service can be updated or accessed. In user authentication, ME relies on the PTD's method to authenticate the user and no extra local authentication method is applied. As the device is assumed to be highly personal, the PTD's authentication mechanism is assumed to be adequate.

Figure 3.3 describes the internal structure of the ME-service. It has two different databases. One of the databases holds personal information that could be delivered to the services and the other holds service definitions which help to authenticate frequently used trusted services.

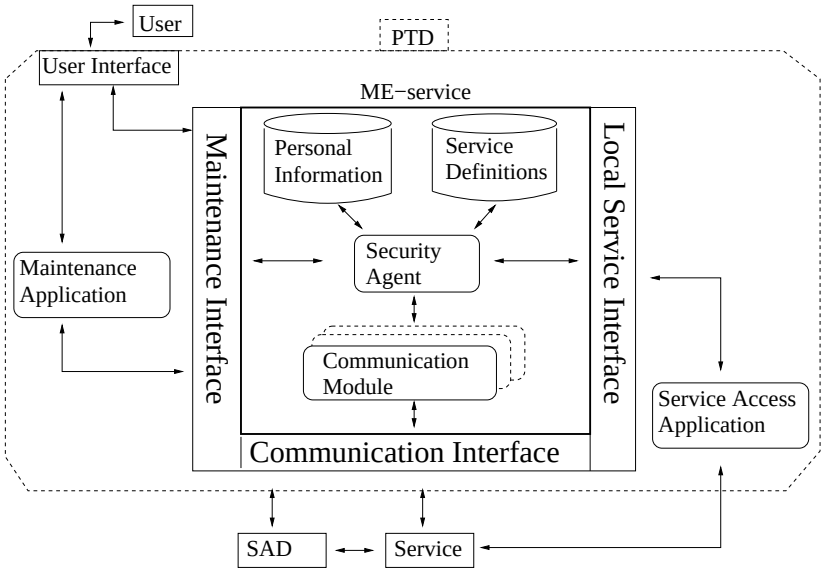

Figure 3.3: ME-architecture

Since there can be different types of communication partners, three different interfaces are required. One interface is needed for communication with the user so that the user can control the service and upkeep of the stored information. Two other interfaces are used for communicating with the partners requiring the personal information. One of them is used by the applications that reside on the same PTD as the ME-service. The other is used to communicate with the service provided by remote devices.

Finally, there are two functional entities in the architecture: the security agent and the communication module. The security agent assures that the information is safe and can be accessed only by authorised services. For various types of communication protocols, there exists a communication module that handles the communication with that given protocol.

3.4.1 Databases and security agent

A personal information database holds all the personal information about the user. The personal information database also contains the security level for each piece of personal information. All of this information is notated in XML based on the XML-schema which was shown in Schema 1 on the preceding page.

The service definitions database holds the names of the services the user has accessed. In this case the name can be a unique name, IP address, Bluetooth Address or something

Algorithm 2 Part of service definitions Schema

```
<xs:element name="Service">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="ServiceName"/>
            <xs:element name="ServicePublicKey"/>
        </xs:sequence>
        <xs:attribute name="SecurityLevel" type="MESecLev"/>
    </xs:complexType>
</xs:element>
```

else that the service transmits as its identity. The name is needed for determining which public key is to be used for encryption. Also, when public keys are transmitted the service name should be the same name that is used on the public keys certificate. For every service there is also defined what security level information it can access. The important piece of the simple XML schema for services is shown in Schema 2.

The *security agent* determines whether or not to give out the requested information, and how the information is secured when delivered. Once the personal information is requested the security agent first checks the security level of each piece of information requested. If the information is very generic, like a favourite colour, it is transmitted to the service right away. If secret information is requested, the security agent uses the service's authentication information and checks how trusted the given service is. If the service's security level is lower than the information's security level, the security agent sends a request to the user through the Maintenance Interface to approve the transaction with the service.

After checking whether the given piece of information is allowed to be transmitted to the service, the security agent determines whether the information has to be encrypted. Since the mobile devices tend to have low computational power it is not feasible to encrypt everything. The decision whether encryption is used or not depends on the security level of the information and the user's definition after which security level the information has to be encrypted. For the actual encryption the security agent creates a symmetric encryption key: $SKey$. The symmetric key is then used to encrypt the personal information. The security agent then encrypts the symmetric key with the service's public key and bundles it all in one message: $E_{ServicePuclicKey}(E_{SKey}), E_{SKey}(PersonalData)$.

Besides determining the information transfer, the security agent also determines whether information can be changed in the databases. The general rule is that only changing requests coming from Maintenance Interface can affect in the content of the databases. Furthermore, before the changes are updated in the database, the security agent requires the user's approval, which relies on the user authentication method of the PTD.

3.4.2 Interfaces

The *Maintenance Interface* is used for upkeeping the information, defining the trustworthiness of various services and communicating with the ME user while the other two interfaces are used for the communication of personal information sharing.

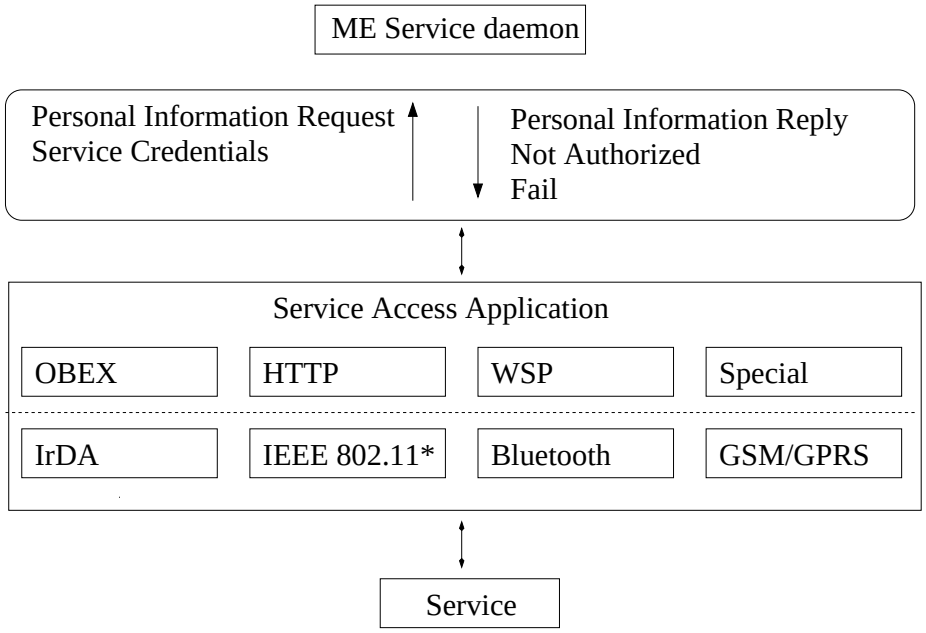

Figure 3.4: Communication with Service Access Application

The *Local Service Interface* is used by the applications on the Personal Trusted Device that require personal information, e.g. when the user connects to the Internet service using a WAP browser (Figure 3.2, case b).

The actual communication with the service is performed by SAA (Figure 3.4). Therefore, the Local Service Interface can be kept quite simple. Different SAAs can support different protocols in service access. A WAP browser might use WSP (Wireless Session Protocol) over Bluetooth while an HTTP browser can run on top of one of the IEEE (Institute of Electrical and Electronics Engineers) 802.11 family WLAN (Wireless Local Area Network) standards. The service's request for personal information should be encapsulated in the existing higher layer service access protocols, such as HTTP. The actual encapsulation of personal information requests is defined in detail on chapter 4.3.1 where applying ME in HTTP based services is discussed.

Once the SAA knows what information the service requests, it can request the personal information from the ME with a *Personal Information Request* message. This message contains the tags of the requested information based on the XML-schema for personal information (Schema 1 on page 35). A Personal Information Request also contains a bit that defines whether the connection between the SAA and the service is encrypted. Some SAAs use encryption automatically when connecting to the service. For example SSL (Secure Sockets Layer) is often used between browser and service when the transfer of private information is anticipated. The encryption bit ensures that the security agent does not encrypt the secret data unnecessarily.

When service authentication is required, the SAA can provide the requested credentials for the service to the security agent with a *Service credentials* message. With the same

message the SAA can also inform if it has already verified the identity of the service. This is usually done when creating an SSL connection.

The ME-service has three different reply messages that can be delivered to SAA: *Personal Information reply*, *Not authorised* or *Fail*. The Personal Information reply contains the requested information in XML. If the connection between the SAA and the service is not encrypted the security agent encrypts the personal information that requires encryption.

The *Not authorised* message is used to inform the SAA that the requesting service is not authorised for accessing the requested data while the *Fail* message tells that the requested data could not be found.

One Personal Information Request can request several pieces of personal data. Therefore, one request can result in several replies as some of the requested data might not exist and for other data the service might not be authorised. The actual implementation of the Local Service Interface should be in the external interface library. The external interface library can then be used for developing a plug-in for the ME that supports the SAA.

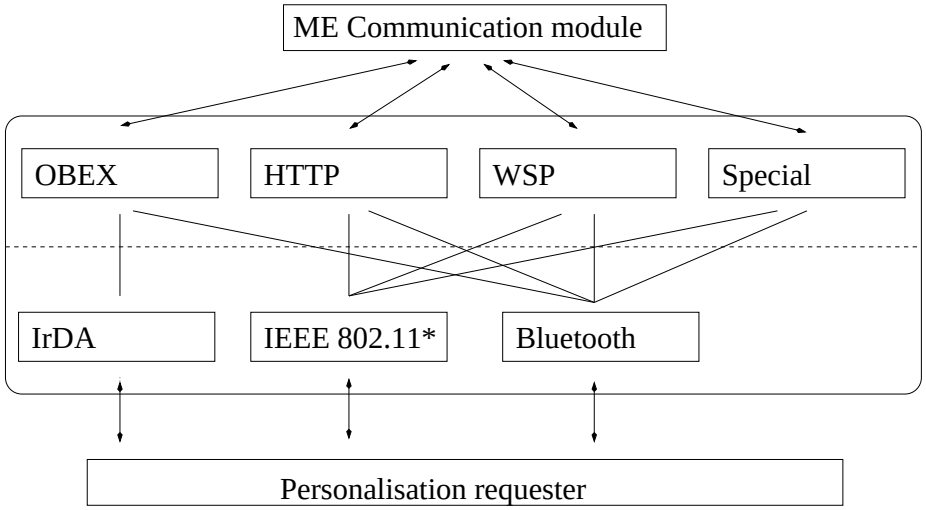

Figure 3.5: Communication through Communication Interface

The *Communication Interface* (Figure 3.5) is used for communication between the ME-service and an external device. The external device can be a device that provides a personalised transparent service such as a personalised augmented reality (Figure 3.2, case a) or it can be a service accessing device (SAD) that requests personal information to be delivered to the Internet service (Figure 3.2, case c). In the first case, the requesting device is controlled by the service provider and the data can be used right away. In the latter case, the device (SAD) is controlled by the user and the program in it will have to transfer the data to the actual service.

The data comes to the Communication Interface over a wireless communication medium in a higher layer protocol message. In a wireless communication medium the ME relies on the capabilities of the PTD. If we think of mobile phones, they all have support for at least one of the mobile telephony technologies such as GSM, GPRS and UMTS

(Universal Mobile Telecommunications System) [58]. Since use of these technologies can become expensive as the service provider charges for every packet delivered, the mobile phones have other wireless communication methods that can be used for short-range communication.

Most of the mobile devices have an IrDA port for short range data communication [59]. Already in the year 2000 there were more than 50 million IrDA units installed and the amount has been growing ever since [60]. IrDA uses an infrared beam for the communication and thus line of sight is required between the PTD and the service/SAD, which might hinder the usability.

The wireless RF-technologies do not require line of sight for communication and thus are more flexible. Laptops and PDAs use Wireless LAN to get a connection to the Internet and its services. The most common standard for the Wireless LAN is the IEEE 802.11b [61]. WLAN support is expected to be implemented also in the new mobile phone models. The IEEE 802.11 family supports also ad hoc networking where mobile hosts can communicate directly to each other. Thus it is feasible for both the transparent service and the SAD approach.

In recent years, Bluetooth wireless technology has emerged in mobile devices [13, 62, 63]. According to Seamus McAteer from Zelos Group, 12% of GSM handsets shipped in 2003 will support Bluetooth wireless technology [64]. Like 802.11, Bluetooth also supports ad hoc networking. The strength of Bluetooth is that the standard itself defines service discovery protocol. Therefore, searching the devices that provide ME-service is robust and requires very little work from the service provider [65, 66]. The Bluetooth standard also contains several profiles which define how the protocol stack should be used in different use cases, such as transferring vCard [67]. The profiles solve the compatibility problems that come through different interpretations of the standards.

The lower layer communication should not directly affect the ME-service and should be handled by the PTD. The ME just informs which kind of communication it supports. The PTD delivers the higher layer protocol messages, such as HTTP or OBEX (OBject EXchange)[59] to the ME-service to the communication interface. The communication interface the delivers the incoming messages to the proper communication module depending on the protocol used.

3.4.3 Communication modules

the ME-service has a communication module for each higher layer protocol it supports. The job of the communication modules is to decode the messages coming through the Communication Interface so that the security agent understands them and to encode the security agent's message to the proper protocol message.

For the communication protocol there are two possible approaches: use of already existing protocols or have a new specialised protocol for personal information transfer. The service provider is likely to have knowledge about an existing protocol and also have support for it on the service providing platform. Existing protocols may require extra protocol layers and provide unnecessary functionality that slows down the communication. They may also lack functionality that would be useful from the ME point of view. Creating a new protocol allows the optimisation of communication for the purposes. On the other

hand, creation of a new protocol would require that service providers add support for it. That would mean more work for service providers and would hinder the utilisation of ME-services. Whether the communication is encapsulated in existing protocols or its own protocol is used, the naming of requests and replies should follow the local interface naming scheme, if possible.

Existing protocols have several candidates, which can be used with little tweaking for ME purposes. SyncML, which is used for synchronising different types of data, supports three different transports: HTTP, WSP and OBEX [53]. In the future versions of the ME SyncML may be used to synchronise data between various databases. Thus the use of HTTP and OBEX as transport protocols is considered here.

Both HTTP and OBEX require a server running at the mobile device. The service requiring personal information connects to the server with the request. The ME-service can implement its own server or it can use an existing server on the PTD, in which case the server has to provide the messages for the communication module. The case where the server is part of the ME-service is described below.

Internet based services are provided over the HTTP protocol. There is even an implementation of the HTTP server that is running on a SIM (Subscriber Identification Module) card [68]. HTTP relies on TCP/IP which will cause an extra package on the communication system. On HTTP, the requested document is defined on the GET message and is in the form of filename?arguments. Thus the possible request for the customer name and address would be as follows:

GET PERSONAL_INFORMATION_REQUEST?<NAME><ADRESS>

The communication module decodes the request and requests the information from the security agent as would be done through the Local Service Interface. When getting the information from the security agent the communication module formats the reply. Replies to the HTTP request result in several unnecessary headers and an HTML (Hypertext Markup Language) document. A simple reply without headers to the previous request could be:

<NAME> PEKKA</NAME> <ADDRESS>LINNUNRATA 1 </ADDRESS>

A lighter approach would be the use of the OBEX protocol. The OBEX protocol has been originally developed by the Infrared Data Association (IrDA) for the transmission of a simple data object, such as vCard, over infrared [59]. Due to the fact that most mobile devices support infrared communication, there also exists lightweight implementations of the OBEX server. OBEX is also supported by Bluetooth wireless technology, which also provides the effective service discovery method.

The use of OBEX and Bluetooth as the Communication Interface is described for transparent services in chapter 4.2 and for SAD in chapter 4.3.2.

3.5 Discussion

The chapter defined the basic architecture for the Mobile E-personality. The current architecture allows the input of the data to ME-databases only through the user interface of the mobile device. This also prevents the approach where the service can use customer

mobile device to store service specific data like in the IDRepository [37]. This possibility has been left out on purpose as it would provide more possibilities for malicious people to burden the customer.

Some research has already been conducted at the Communications Engineering Laboratory of Lappeenranta University of Technology on allowing the use of a desktop computer in upkeeping the data stored in a ME by utilising SyncML. In the future, this option as well as the possibility for services to provide service-specific information to be stored in a ME may be added. SyncML supports also enable the possibility for backing up the data on a personal computer.

These additions require an evaluation of the risks they will bring. Based on the evaluation, improvements to the current version's security mechanisms should be done. Additional security mechanisms, such as transaction logs, could also be considered. The benefits and drawbacks of such mechanisms in a mobile environment should be carefully considered.

CHAPTER IV

ME use cases

The basic description of architecture does not tell how the described architecture works in different cases. This chapter presents different cases and describes how the ME can be utilised in them. The approach is similar to that used in the Bluetooth standard. The Bluetooth standard is divided into two parts, where the first part describes the technology and the second part contains profiles which define how Bluetooth is used in different cases.

The first section describes how authentication works with the ME. Both service authentication to the ME and customer authentication to the service are explained. After authentication the utilisation of the ME by transparent services is explained.

4.1 Authentication

Authentication is needed to prove that the communicating participants are who they claim to be. When using services the parties that might be authenticated are the service and the user. From the ME point of view there are two cases when the authentication is needed:

1. The user has to authenticate himself to the PTD and ME to ensure that only the owner of the PTD can upkeep the personal information and start the running of the ME-service.
2. When providing a subscription service, it is important to authenticate the customer as a subscriber.

On the other hand, simple personalisation of the service does not require validation of the identity of the customer. Thus before implementing the customer authentication, one should carefully consider whether the authentication is actually needed or not.

Service authentication is required before confidential personal information is delivered. As was stated in the previous chapter, in some cases the Service Access Application

can do the service authentication on behalf of the ME-service. Similarly, when using a trusted Service Accessing Device such as a desktop computer at home, the application in the SAD can do the service authentication on behalf of the ME-service. In this case, the ME-service should check the authenticity of the SAD.

4.1.1 User authentication

The ME-service relies on the PTD for the authentication of the user. Therefore, the user authentication methods are discussed here from the service and service provider point of view.

Authentication of the user can be based on three different aspects: something the user knows (password), something the user is (fingerprint) and something the user has (key). After a brief browsing of the Internet, it becomes quite clear that passwords are where the services on the Internet rely on. Unfortunately there are several problems in the plain password-based authentication. The biggest problem is that users tend to choose simple passwords. The passwords are based on names or words, which are easy to crack with a dictionary attack. Simple passwords are used because they are easy to remember [69, 70]. One of the most known password thefts was done by Ali Baba when he eavesdropped on the captain of the thieves saying "Open sesame" to get into the magical cave in the story of Ali Baba and forty thieves in Arabian Nights [71].

When using more services, the customer has to remember more passwords if he has different passwords for every service. Therefore, some customers use the same password for different services. A malicious service provider can store the password that the customer uses for his service when the user registers. The registration form may enquire what other services the customer uses. After the customer has filled in the registration form, the service provider has a password and a number of services to try it on. Using different passwords for different services does not solve the problem entirely either. The more there are passwords the higher is the possibility of typing the wrong password to the service. A malicious service provider can store these wrong passwords in a database along with the username and then try to use these in various services [31].

Web browsers can help with the password remembering problem. Just like they can store user's personal information they can also store the password and the URL (Uniform Resource Locator) where the password was used. When the user is asked the password at a site he visits, the browser fills in both the username and the password automatically. The user just has to accept the transmission of the password. This approach relies heavily on the security measures of the SAD. For example if it is used on the computer running Windows 95, basically anyone with some knowledge can authenticate himself to the services as anyone who has stored service passwords in the browser. Furthermore, the problem for a nomadic user is the same as in personal information storage: the browser at home does not know the password for the service you have registered at from the work.

To avoid multiple passports, third party based single sign-on approaches can be used. The problems with these approaches are the same as with personal information storage: Who trusts who? Is there a connection available to the third party? What does it cost? and so on.

Using biometrics for authentication over a network possesses several problems. First of all, there is a need for a biometric reader. Secondly, once the biometric data has been read, it has to be sent over the network. Thus the service provider gets the data and can use it to authenticate himself as the customer to other service. A similar problem exists when using an authentication device like iButton [72]. While in iButton, the authentication is done so that no secret is transmitted over the network; the user has to share another secret with the service. An external reader is also needed for iButton.

If the ME-service is wanted for user authentication, the method should not require any external readers. There should be no secret passwords transmitted over an unencrypted communication medium. In order for the system to work with many different services there should be no shared long time secret involved in authentication. The answer to these demands is a challenge-response authentication protocol that uses asymmetric cryptography.

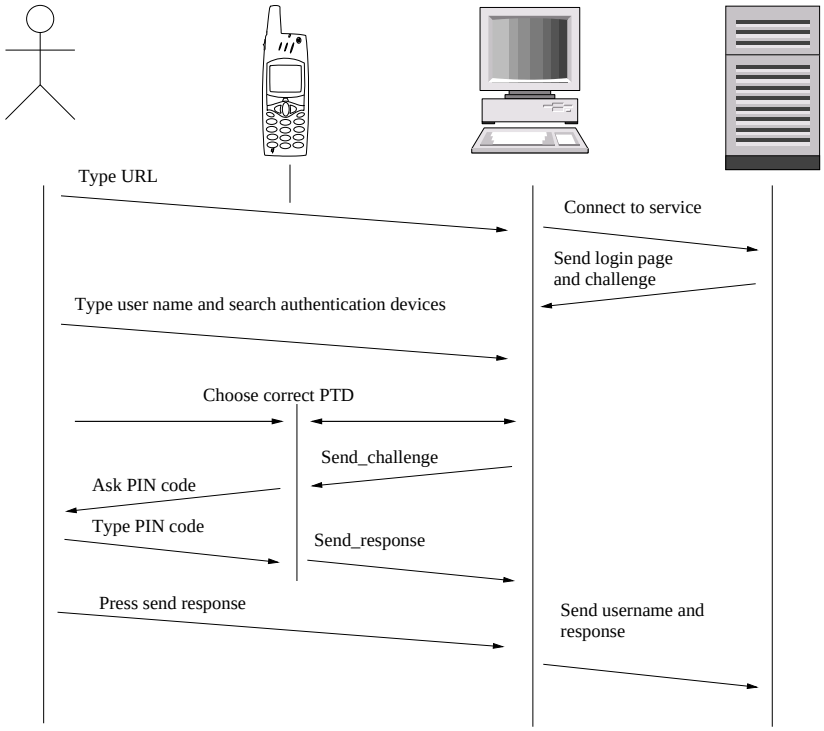

Figure 4.1: User authentication to service.

There are several existing authentication solutions that rely on challenge-response authentication with asymmetric cryptography. For example, the Finnish government endorses an electronic ID card approach called FINEID (FINnish Electronic IDentification). The FINEID card holds the user's identity and asymmetric keys certified by the Finnish government. The card can be used by the user to authenticate himself to the services supporting FINEID based authentication. Another requirement is a smart card reader on the computer that is used for authentication [73]. The latter requirement can cause

problems, especially when services are accessed with a personal trusted device.

When the user connects to the service first time, his public key is delivered among the other registration information required. The service stores this public key to its database with the pseudonym the user has provided as an account name. The next time the user wants to use the service (Figure 4.1), the service creates a random bit-stream as a challenge. The service then encrypts the challenge with user's public key and sends it to the user. The challenge is delivered to the ME-service as a request of <Authentication_Response> personal information. The security agent does not look for authentication_response data from the database but instead requests the user's approval for sending the response to the challenge. If approval is given, the security agent uses the stored secret key and decodes the challenge. The decoded challenge is then encoded with the services public key that lies in the service definitions database and is then delivered back to the service as <Authentication_Response> personal information. Now the service has only to know the user's public key. Therefore no secret is shared and the same key can be shared with any service.

A more detailed process for service authentication when using PTD is described in *Publication V.*

4.1.2 Service authentication

The more automated the service use is the more important it is to authenticate the service. The email system already has huge problem with spam [74]. Today finding the real messages from the mailboxes is a cumbersome task. If we want to have automated messages popping on the screen of a mobile device, there has to be a way to decide who can provide these messages and then the provider has to be authenticated. This is discussed in *Publication VI.*

For the ME, service authentication is necessary in order to make the decision when some information can be sent without the user's explicit action of approval. For Internet-based services which are accessed with SAA, the problem can be solved by using SSL. SAA uses SSL to authenticate the service and then transmits the public key of the service to the ME.

In transparent services that are provided over Bluetooth, SSL is a bit too heavy approach. The problem setting is quite similar to the one defined in *Publication VI.* If we think for example that International Coffee Shop has several coffee shops around the world, they all can't share the same asymmetric key pair. On the other hand, it would be pointless for a customer to store the public keys of all the shops. Therefore, we need to use certificates.

The idea of a certificate is that a trusted party becomes a certificate authority (CA). In the case of International Coffee Shop, it could be their international headquarters. The ME-service has stored the international headquarters' public key (Figure 4.2 CA) and the security definitions base on that. The international headquarters then signs the public keys of all the national headquarters (Figure 4.2, D1). The signing of the public key denotes they guarantee that they trust the national people following their standard of security. Finally, the national headquarters signs the public keys for the cafeterias in their own areaa(Figure 4.2, D1).

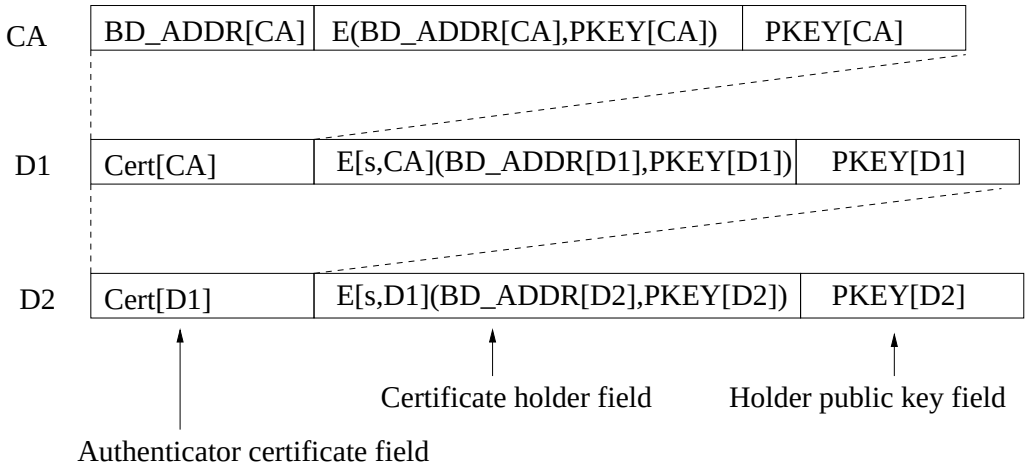

Figure 4.2: Certificate tree

When the customer arrives at a local cafeteria, the cafeteria service sends the personal information request along with the certificate containing its credentials. The security agent first verifies that the certificate is actually certified by the headquarters. This is done by verifying the signatures. Bluetooth device address (BDADDR) can be then used to validate that the certificate belongs to the party that provides it. This is just an extra measure and is not necessary as the final check comes from the encryption of the personal information. Only the owner of the secret key corresponding to the certified public key can decrypt the symmetric key that is needed to decrypt the transmitted personal information.

4.2 ME and transparent services

Transparent services are the services to which accessing requires no action from the user. The service can be automated light adjustment when the customer enters the room or a special greeting in the customer's Bluetooth head-set when he enters the shop. In this section, how transparent services can use ME to provide personalised service is presented.

From the various possibilities discussed in the case of the communication interface in chapter 3.4.2, Bluetooth is used here for transparent services. For service providing there exist several Bluetooth access point manufacturers. There are also some concepts developed for providing services to a wider area through several access points. One such product concept is Ericsson's BlipNet [75]. The existing products provide a good basis for developing new services.

From all the higher layer protocols that Bluetooth supports, OBEX is used here and therefore Generic Object Exchange Profile (GOEP) from the Bluetooth standard is followed. BlipNet also supports OBEX.

For the use of OBEX and Bluetooth for communication between the ME and transparent services, ME has to run the OBEX server. In order for the ME-service to be searchable

by transparent services, the ME-service has to be registered in the PTD's Bluetooth services. Furthermore, the user's PTD has to be in General Discovery Mode and the *ME-service* has to be up and running.

The transparent service providing for example augmented reality constantly searches devices in its communication range that have the capability to provide personalisation information, i.e. devices that run the ME-service. This is done by using the Service Discovery Protocol (SDP) from the Bluetooth standard. The use of the Service Discovery Protocol for such a task is defined in the Service Discovery Application Profile [66, 67]. When the user arrives in the area of the transparent service, the user's PTD responds to the services SDP query by telling the PTD's capabilities, i.e. informing that the PTD has a ME-service from which personal information can be requested. The transparent service then creates an OBEX connection to the PTD by following the definitions of GOEP from the Bluetooth standard.

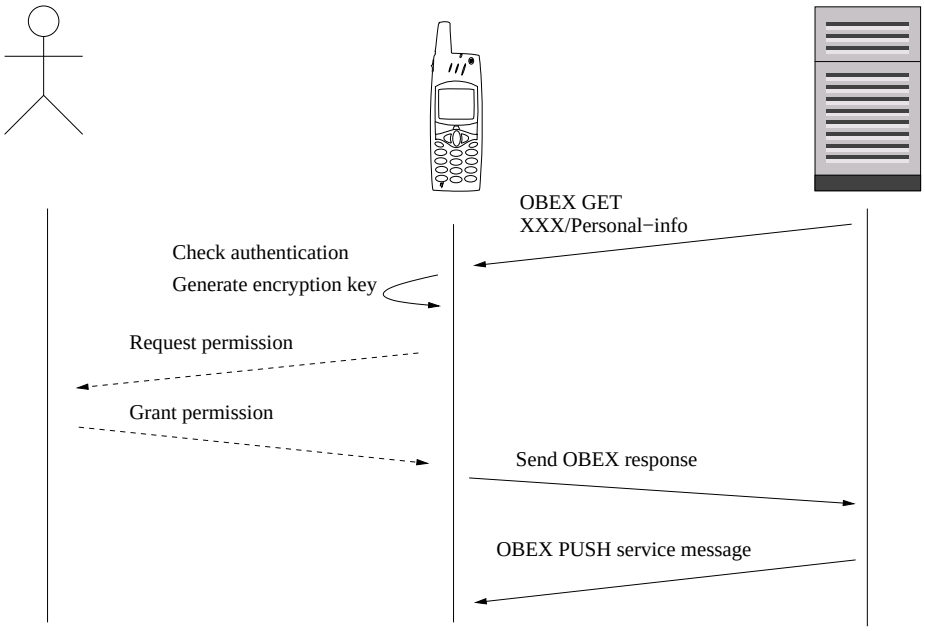

Figure 4.3: Personal information transfer to transparent service

Once the OBEX connection is created, the actual personal information requesting can be done. The communicating is described in Figure 4.3. The personal information request should be encoded in OBEX GET message. The OBEX GET message has a *type field* which defines the MIME (Multi-Purpose Internet Mail Extensions) type of the requested information. For the MIME type we can use XXX/PERSONAL-INFO.

The ME-service has to be told which pieces of personal data are requested. In addition, when private information is requested the service credentials for authentication purposes have to be delivered. All this information can be encoded in the type field of the OBEX GET message.

The different pieces of data are separated by colons, i.e. the type field is as follows:

XXX/PERSONAL-INFO: SERVICEINFO: REQUESTED-DATA. The SERVICEINFO consists of a service identity and possibly a X.509-based certificate. The REQUESTED-DATA field contains the information the personal information service wants from the user, based on the XML-schema which was described in Schema 1 on page 35.

When the GET message reaches the ME, the request goes to the security agent, which first authenticates the services as stated in subchapter 4.1.2 on page 46. Then security agent decides what data will be sent and what needs user approval before sending. The security agent then encrypts all the provided information to one object and sends it all in an OBEX response message along with the asymmetrically encrypted symmetric key.

In case the service does not have a public key provided in the certificate and the transferred data should be encrypted, the ENCRYPTIONKEY field can be added at the end of the MIME type definition. This new field contains information for encryption key exchange that is done by using Diffie-Hellman key-exchange [76].

In Diffie-Hellman key-exchange, the transparent service generates values g, n and y. It calculates $Y = g^y mod\, n$ and sends g, n and Y in ENCRYPTIONKEY field to the PTD. PTD generates value x and calculates $X = g^x \, mod\, n$ and $k = Y^x \, mod\, n$. Value k is used then to encrypt the personal data before it is sent to the service along with value X [76].

The use of Bluetooth for transparent services raises a special concern for the user's privacy. If personalised services are used, it means that the mobile device has to be discoverable for the service. Therefore, the service can access the device's unique Bluetooth address. A malicious service provider can use this information and the unique identifier to create a database about the data learned from the customer, even though the customer prevents any other identifying information transfer.

4.3 ME and indirect service access

In indirect service access, the service and the PTD do not communicate directly with each other. Instead, there is a separate service accessing device on which the service accessing application acts as an intermediary for personal information transfer. Due to the external device, the service cannot request the personal information directly from the PTD but has to rely on the SAD. Therefore, the communication model can be divided into two separate communication channels: *Service-SAD* and *SAD-PTD*.

Indirect access is used mainly by nomadic users to access Internet services from a variety of locations. Therefore, it is assumed here that HTTP is used as a service access protocol between the service and the SAD. The following approach can be translated for other protocols if required.

Communication between the SAD and the PTD will be independent from the service access protocol. Thus the defined the SAD-PTD communication model can be used even if the service access method from the SAD changes, i.e. other than an HTTP based service is used. In SAD, the SAD-PTD communication is handled by a plug-in program created for the service accessing application, e.g. a Web browser. The plug-in uses the ME communication interface and the method defined for personal information transfer between the mobile device and the transparent service described in the previous

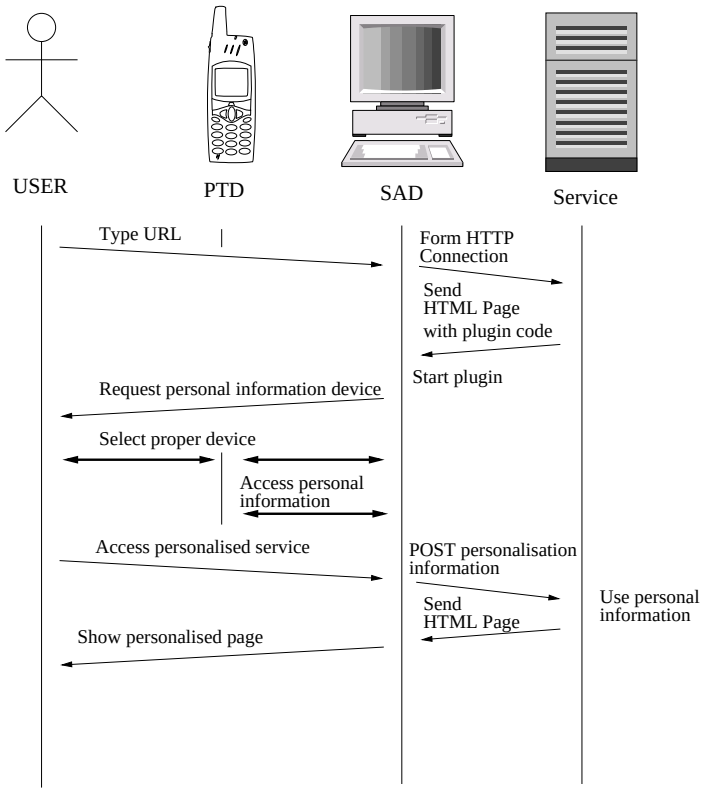

Figure 4.4: Generic HTTP-based service access

chapter. With the use of plug-in, there is no need to modify the actual service accessing application.

The general use of the indirect HTTP based service which requires a personal information request from the mobile device is described on Figure 4.4. The use of the service starts when the user types in the URL of the service in the browser running on the SAD. The browser then sends a request to the service to get the initial Web page of the service.

The initial Web page contains among normal HTML definitions a small JavaScript code that starts the browser plug-in code. This plug-in handles the Bluetooth based communication between the SAD and the PTD which is described in *Publication III.* Once the plug-in is loaded it forms a Bluetooth connection to the PTD. The Bluetooth address of the PTD can be preset for the plug-in on the personal computer. If the PTD's address is not set (i.e. when using for example a computer in the library), the plug-in shows a list of Bluetooth enabling devices on its perimeter. From this list, the user has to choose the device acting as the PTD.

Once the connection has been created the plug-in requests the personal information from the SAD. The plug-in then inserts the information it receives into the appropriate fields on an HTML document. If all the requested information was not found in the PTD, the user can fill in the empty fields by hand. The document is then delivered back to the

service when the user clicks the *personalise* button of the document.

4.3.1 Service-SAD communication over HTTP

Communication between the service and the SAD is done by using the HTTP protocol. The messages needed for the personal information access are encapsulated in the payload of HTTP messages. This requires additional functionality to the SAD as well as in the service but allows the customer to use an already familiar way to access the services.

On the server side, the page requiring personal information has to include a personal information request in it. Also the server side has to be able to read the incoming personal information. The reading functionality is handled by the server side script.

In the SAD, the SAA, e.g. browser, has to be able to extract the personal information request from the incoming Web page, communicate with the ME and finally transmit the personal information to the service. This functionality is handled by additional browser plug-in software.

Before the plug-in can work, it has to be activated. Therefore the web page containing personal information request also has to have javascript that activates the plug-in software. The actual personal information request can be inserted in the document in two different ways. It can be either a parameter for the javascript that activates the plug-in or it can be part of the document.

In the HTML document approach, the request is encoded on the Web page in distinguished field that's ID is PERSONALISATION REQUEST. Plug-in can find this field by browsing the DOM (Document Object Model) tree of HTML document. The field contains a list of personalisation information that the service requests. If the HTML Document holds a form to be filled in, the field names on the form should be notated following the XML-schema for personalisation information (Schema 1 on page 35). This way the plug-in can easily check the requested information from the field names. If the request is delivered as a parameter the request can be gotten right away by the plug-in, which is faster than browsing through the DOM tree.

No matter which way the request is got by the plug-in, it consists of the list of the wanted personal information and optionally the certificate of the service. This information is then forwarded to the PTD as described in the next subchapter.

The response to the service is encapsulated also to the payload of an existing protocol message, such as HTTP POST. The response can consists of the requested information as a whole, a special code for providing a reason for failure to get information or part of the information and an explanation code. Again, the personal information is tagged in XML following the XML-schema for the personal information and is part of the response document.

4.3.2 SAD-PTD communication using OBEX

The communication between the SAD and the PTD is done over the communication interface of the ME-service and thus resembles the communication of a transparent service and the PTD. It relies on OBEX over Bluetooth and has two separate steps. The first

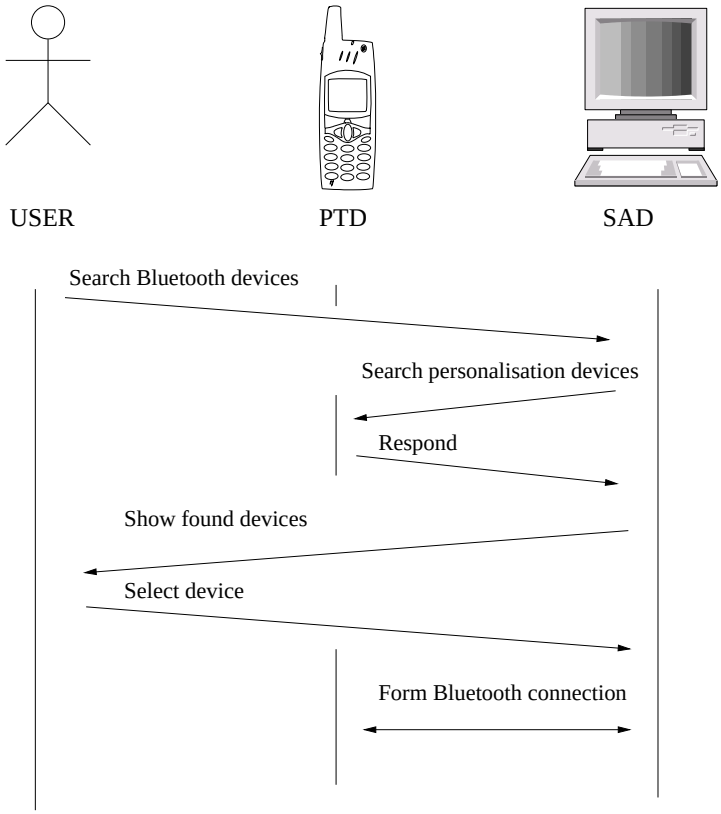

Figure 4.5: Communication channel forming between SAD and PTD using Bluetooth

step is to discover the Bluetooth devices that have a ME-service and form a connection to that correct device (Figure 4.5). After this, the retrieval of the personal information from the PTD can be done (Figure 4.6).

The discovery and forming the connection differ a bit depending on whether a *trusted* or a *foreign* SAD is used. A trusted SAD is a device that is used frequently and can be thought of as personal equipment, e.g. a personal laptop or a dedicated workstation at the office. A foreign SAD is a device that is used by various people that the user doesn't necessarily know, e.g. a computer at a coffee shop or a public library. On a trusted SAD the identity of the PTD is known by the plug-in and thus the Bluetooth connection can be made directly without using slow discovery procedures. On a foreign SAD, the plug-in software has to use Bluetooth SDP first to find devices that provide the ME-service. The SAA then shows all the devices found in its window from where the user can select his own PTD. After the correct Bluetooth device is determined, the normal forming of the Bluetooth connection takes place. Once the connection is formed between the SAD and the PTD, the PTD acts as an OBEX server and the SAD as an OBEX client. The SAD can now request personal information from the PTD.

In the actual communication between the SAD and the PTD, the type of the SAD can

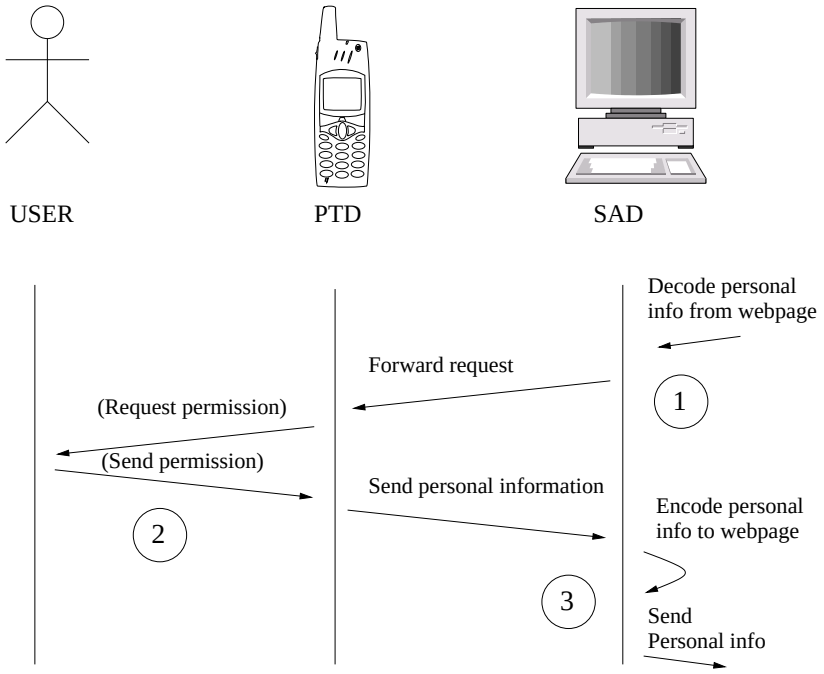

Figure 4.6: Personal information retrieval from PTD by SAD

be also taken into account. When using a trusted SAD, it is possible to let the SAD do the authentication of the service as well as encryption of transmitted data. In the case of a foreign SAD, the PTD has to do all the work. A trusted SAD is recognized by its unique Bluetooth device address (BDADDR). Even though the BDADDR can be forged, the authentication method can be thought adequate for the given purpose. A malicious SAD creator should first know what BDADDRs are defined as trusted ones by the given user. Thus the attack would require knowledge about the user and also a need to get him to use the malicious SAD.

To allow the PTD to make the service authentication and encryption, the SAD has to provide some information about the used service. A trusted SAD can also provide information if the service is authenticated by the SAD and whether the connection between the SAD and the service is encrypted or not.

These pieces of information are encapsulated in the OBEX message along with the list of requested personal information. Thus the personal information request on OBEX GET consists of 2 fields: Serviceinfo and requested personal information fields. As in transparent services, these fields are separated from each other by a colon: XXX/PERSONAL-INFO: SERVICEINFO:REQUESTED-DATA. The authentication and encryption level are included in the SERVICEINFO field.

SERVICEINFO contains the identity of the service and either a certificate of service or the service's public key. The plain service public key is provided only when the trusted SAD has performed the service authentication. In this case the authentication level is set to 1. In all other cases the certificate is provided. The encryption level defines the size of

the symmetric encryption key used in the SAD-service connection. If no encryption is used the encryption level is set to 0. In practise this means that when SSL is used for service access, authentication is set to 1 and the encryption level is bigger than zero.

For the personal information request, the PTD has two types of responses: *personalisation response*, which contains the requested personal information and *error response*, which tells the reason why the requested information was not delivered. The reply is encoded as one object, which is sent to SAD in OBEX response message.

4.4 Discussion

The cases described in this chapter show that a different approach for personal information storage provided by a ME-service is feasible. At the same time the ME also opens up new possibilities for the use of personal information and mobile devices. For example, fitness devices can acquire information about the user and provide customised practise guidance [77]. These new opportunities will affect the further development of the ME. The communication method, for example, may require additional development to allow more sophisticated and flexible communication required by new types of services. Instead of being just a passive information source the ME may be developed to be active by adding some intelligence to it in the form of context awareness. The current version of the ME provides a solid base for further development.

CHAPTER V

Future work

The architecture presented in this thesis provides a solid base for further development of the system. The main emphasis of the thesis has been on the communication between various types of services, leaving several other issues to be addressed in the future research.

For the ME to be useful outside the laboratory environment, a standardised markup is required for personal information. The notation used in the ME should be further developed to take into account more than just basic information. The basic name and address are not enough, but a variety of user's personal preferences should be included. This requires feedback from the user modelling research community.

In the current version, the information is stored in the mobile device's memory, but there are also other possibilities. One viable possibility is to use a SIM card, which has already been used in different ways, such as in the HTTP server mentioned earlier. The SIM card approach can provide extra security for the data, but at same time restricts the amount of devices that can provide the ME-service.

As the mobile device may have several applications that want to authenticate the accessed service, the service authentication could be separated from the ME as an external service. A single service authentication module can save valuable storage space on a mobile device. It is also more convenient for the user as he has to upkeep only one service credential database.

Also the security model of the ME needs also further research. Addition of timestamps for personal information as well as time of validity can add extra security but may also hinder the usability. Logging of transactions can help reveal misuse. Unfortunately, logs require storage space, which is not abundant in mobile devices. The current security levels based approach for deciding whether the information is transmitted or not is just a slight improvement to the policy where the permission is asked from the user all the time or that all information is disclosed to all services. Further on the security levels for stored information should be defined so that the access to information is easy to handle and understand. For example P3P (Platform for Privacy Preferences) used in the

World Wide Web sounds good in theory but is too complicated for the average user to understand [78, 79].

One interesting direction is to develop the ME to be a PeerHood service in the mobile device [80]. PeerHood provides the basic communication capabilities over a variety of networking technologies. As PeerHood also includes robust service discovery, the ME-service can be found regardless of the communication technology. Therefore, the ME itself can be independent of the communication medium.

The current model of just providing information to services can be changed so that services can also store information in the ME. For this purpose, the SyncML standard should be taken into account. Through the two-way communication, new types of services and new ways of using mobile data can be developed.

Inputting and handling personal information in the mobile device can be rather tedious and annoying. To improve the usability of the ME, user interface development is required. With the plain text editing interface of raw data, no-one is going to adopt the system. With SyncML it is also possible to handle the data on a desktop computer and then synchronise it with mobile device.

Adoption of the ME to the real world requires co-operation of several parties from big service provider companies to standardisation organisations and mobile device manufacturers. How the ME will affect especially the service providers' business models is something that has to be researched in order to get them to support the development. The effect can be more significant in areas where companies sell the customer data to each other. Finally, it is important to research how the ME relates to the privacy laws around the world.

Chapter VI

Conclusions

In this thesis an approach where the user's personal information would be stored in his Personal Trusted Device (PTD) is presented. The storage of information in a mobile device has no use unless it can be accessed. Therefore, the Mobile E-Personality (ME) has been defined to provide the stored information to those requesting it.

The ME approach provides several advantages for personal information handling. The personal information stored in the ME is accessible to all types of services whenever and wherever the user so desires. The high accessible personal information improves the service's usability by:

- Allowing different types of services to be adapted based on customer preferences, i.e. personalised. Since the information is stored in a single accessible device, it also provides a possibility for the service to be personalised even for the first time user of the service, i.e. avoiding the "cold-start" problem.
- Providing the possibility for anonymous personalisation. With the use of the ME, personalised service can be provided without registration and thus the user's identity can be kept secret.
- Ease the registration process by removing the need for typing in the repetitive information. Information can be used for filling out different types of registration forms, whether the form is provided through the Internet or ubiquitously at the hotel lobby.

High accessibility also provides new opportunities for service providers. Personalisation can be implemented in new types of services. For example, ambient environments have a way of finding out the preferences of the people inside. The advantage, in terms of personalisation, of huge customer databases that the big companies have is reduced. Small and new service providers have a better possibility to provide personalised services as the customers are more likely to provide their information when it can be done easily.

As the required data can be easily requested when required, the size of the customer databases can be reduced.

Besides the improved usability, the ME provides the user better control over his own personal information. As the information is stored in the user's device, it is also easy to upkeep. The user can define which services can access which piece of information thus allowing secure automation of information transfer.

The security of personal information is improved by keeping the personal information stored in only one place. As the PTD holds information about only one person, it is a less attractive target for malicious people than huge databases.

Hopefully the ME will someday move from the research table to be used by real people and real services. Only time will tell. As the Nobel Prize winning physicist Denis Gabor remarked in his book written forty years ago, "The future cannot be predicted; but futures can be invented."

CHAPTER VII

Summary of the Publications

This thesis consists of six publications from the area of services and personal information. Publications I and II concentrate on aspects of personal information and its storing. Publications III and IV concentrate on mobile device and internet services. Publication III defines the way to transfer personal information to internet service while Publication IV concentrates on mobile device use for user authentication purposes. Publications V and VI concentrate on transparent services. Publication V defines the way to transfer personal information to transparent service while Publication VI concentrates on service authentication so that user can select who can provide transparent services.

Publication I describes various attributes of the personal information and the effect of these attributes to optimal personal information storage location are analysed. The storage location considered were: *Service provider*, *trusted third party* and *the user*. All the given locations had their positive and negative sides. From the user point of view, the best place would be on himself while the services prefer to have it on their databases. Third party is a compromise between the two.

The analysis on personal information was divided in three groups: *properties*, *dependencies* and *ways of use*. Properties affecting to the personal information location were: Stability, size, generality, origin and privacy. Dependencies affected to the stability of the information i.e. the dependant information changed when certain condition changed. Therefore the dependant information would be logically saved on user as user can define which dependencies are affecting at the given time. The information could be used directly like in registration or indirectly to assume user preference based on the information pieces known.

As a conclusion it was realised that most of the information should be stored on user's mobile device to provide user better control on his personal information. Services should store only the information they generate themselves and need when user is not using the service. It was also noted that mobile device as storage place requires definitions how the information is transferred and notated.

The author analysed the personal information and wrote on most parts of the paper

Publication II describes the general architecture of Mobile Electronic Personality (ME). The publication defines the various interfaces for accessing the information stored on PTD. These interfaces are *maintenance interface* through which the personal information is kept up, l*ocal service interface* which is used by the applications on the PTD and *communication interface* which is used by external devices.

The publication also defines the basic principles how the personal information is handled i.e. how to define which services may access the information and which do not. The author developed the architecture and wrote the paper.

Publication III defines how to transfer information stored on mobile device to internet service when the service is accessed with external service accessing device such as desktop computer. The communication is divided in two different parts: SAD-service and SAD-mobile device.

The communication between SAD and service is carried out with HTTP protocol. The webpage requesting personal information contains javascript that launches communication plugin. This plugin decodes the personal information request from the document and handles the communication to mobile device.

The communication between mobile device and SAD is done by using Bluetooth wireless technology and OBEX protocol. The mapping of the personal information request inside the OBEX messages is defined and the actual steps in forming communication channel and transferring data is described. The author designed the communication architecture and wrote generic parts of the paper.

Publication IV describes CRAB (Challenge-Response Authentication over Bluetooth) architecture, which defines how PTD can be used for user authentication on service in the internet. The basic assumption is that PTD plays important role in service use as a source of information, so no new device is required for authentication purposes.

In the given approach the service is accessed by using web browser from desktop computer. When authentication is required, special browser plug-in software forms connection to the PTD by using Bluetooth wireless technology.

CRAB relies on asymmetric cryptography and challenge-response authentication protocol. These were chosen to avoid transferring secrets over unreliable wireless communication medium. There is also need for only one secret as that secret is never shared with anyone.

The main result of the publication is definition how PTD can be used also for customer authentication to the service. The author developed the architecture and wrote the paper.

Publication V concentrates on the question how to transfer information stored on Mobile device to the transparent service. The paper identifies the problem of acquiring personal information to transparent service. Due the transparent and automated nature of the service the information cannot be requested from the user himself. As the transparent services are also usually provided just locally, the information cannot be acquired before hand.

The approach used is *personal information transfer service* running on user's mobile device. The service is provided over Bluetooth wireless technology and relies on OBEX

protocol. The publication also defines three variables of privacy: *service authentication*, *encryption* and *user action*. These variables values can be defined either for piece of information or certain service. For data encryption a protocol for key exchange was defined to keep communication minimized.

The result of paper is definition how transparent service can request personal information from user's mobile device, so that user has some control on his own privacy. The author developed the architecture and wrote the paper

Publication VI Flash Notes over Bluetooth Wireless Technology, describes architecture for providing informative messages i.e. flash notes to the user over Bluetooth wireless technology. The flash note is simple transparent service where the service is provided without user action. The problem with such a service is that user might get service he does not want to. Therefore the main focus of this publication is to define how user can choose who can provide the notes to the mobile terminal and who cannot. This is done by choosing trusted service providers and using certificates to authenticate the source of note. With certificates it is possible for one service provider to provide his services over various access points. The author developed the architecture and wrote the paper.

[1] A. G. Bell, "Letter written by Alexander Graham Bell to his Father, Alexander Melville Bell," 10 Mar. 1876.

[2] C. Gbaguidi, J.-P. Hubaux, G. Pacifici, and A. N. Tantawi, "Integration of Internet and telecommunications: An Architecture for Hybrid Services," *IEEE Journal on Selected Areas in Communications, Special issue on Service Enabling Technologies for Networked Multimedia Systems*, vol. 17, pp. 1563–1579, Sept. 1999.

[3] K. Ikkelä, M. Myllynen, J. Heinänen, and O. Martikainen, "4G mobile Network Architecture," in *Emerging Personal Wireless Communications* (O. Martikainen, J. Porras, and J. Hyvärinen, eds.), (Lappeenranta, Finland), pp. 183–196, IFIP, Kluwer Academic Publisher, Aug. 2001.

[4] K. Ikkelä, "Local Services in a Fourth Generation Mobile Network," Master's thesis, Lappeenranta University of Technology, 2001.

[5] O. Martikainen, V. Naumov, and D. Kolesnikov, "Internet Service Management," in *Telecommunication Network Intelligence*, (Vienna, Austria), IFIP, Kluwer, Sept. 2000.

[6] J. Angel, "Look Ma, No Cables," *Network Magazine*, vol. 15, pp. 42–52, Nov. 2002.

[7] L. Kleinrock, "Nomadic Computing - an opportunity," *Computer Communication Review*, vol. 25, pp. 36–40, Jan. 1995.

[8] J. Ikonen and J. Oksanen, "Wireless LANs and Regional Networks," in *Emerging Personal Wireless Communications* (O. Martikainen, J. Porras, and J. Hyvärinen, eds.), (Lappeenranta, Finland), pp. 197–208, IFIP, Kluwer Academic Publisher, Aug. 2001.

[9] M. Juutilainen, J. Ikonen, and J. Porras, "Connecting Multiple Operators to A Regional Network," in *Proceedings of the IASTED International Conference on Wireless and Optical Communication*, (Banff, Canada), pp. 545–550, July 2002.

[10] J. Hjelm, *Designing Wireless Information Services.* John Wiley & Sons, Inc, 2000.

[11] B. Kasanoff, *Making It Personal.* Perseus Publishing, 2001.

[12] H. Oinas-Kukkonen and V. Kurkela, "Developing Successful Mobile Applications," in *Proceedings on IASTED International Conference on Computer Science and Technology*, (Cancun, Mexico), pp. 50–54, ACTA Press, 19 May 2003.

[13] Y.-B. Lin and I. Chlamtac, *Wireless and Mobile Network Architectures*. John Wiley & Sons, Inc, 2001.

[14] D. S. Bennahum, "Be Here Now," *Wired Magazine*, vol. 9, pp. 159–163, Nov. 2001.

[15] M. Frodigh, P. Johansson, and P. Larsson, "Wireless ad hoc networking - The art of networking without network," *Ericsson review*, no. 4, pp. 248–263, 2000.

[16] C. Toh, *Ad Hoc Mobile Wireless Networks: Protocols and Systems*. Prentice Hall, 2001.

[17] U. Hansmann, L. Merk, M. S. Nicklous, and T. Stober, *Pervasive Computing Handbook*. Springer-Verlag, 2001.

[18] J. Burkhardt, H. Henn, S. Hepper, K. Rintdorff, and T. Schäck, *Pervasive Computing: Technology and Architecture of Mobile Internet Applications*. Pearson Education limited, 2002.

[19] A. Sousa, C. Baquero, J. O. Pereira, R. Oliveira, and F. Moura, "A Human Centered Perspective for Mobile Information Sharing and Delivery," in *Proceedings on ECOOP'96: Workshop on Mobility and Replication*, (Linz, Austria), July 1996.

[20] D. Kammer, G. McNutt, B. Sense, and J. Bray, *Bluetooth Application Developer's Guide*. Syngress Publishing Inc., 2002.

[21] Nokia, "Personal Information Management." http://www.nokia.com/nfb/personal info.html. Accessed January 26, 2004.

[22] R. Hunter, *World Without Secrets: business, crime, and privacy in the age of ubiquitous computing*. John Wiley & Sons, Inc, 2002.

[23] L. Ardissono and A. Goy, "Tailoring the Interaction with Users in Web Store," *User Modeling and User-Adapted Interaction*, vol. 10, no. 4, pp. 251 – 303, 2000.

[24] L. F. Cranor, "'I didn't buy it for myself' Privacy and Ecommerce Personalization," in *Proceedings of the 2nd ACM Workshop on Privacy in the Electronic Society*, (Washington, USA), 30 Oct. 2003.

[25] M. Koch, "Global Identity Management to Boost personalization," in *Proceedings of 9th Research Symposium on Emerging Electronic Markets*, (Basel, Switzerland), pp. 137–147, Sept. 2002.

[26] I. A. Goldberg, *A Pseudonymous Communications Infrastructure for the Internet*. PhD thesis, University of California at Berkeley, 2000.

[27] R. LaRose and N. Rifon, "Your Privacy is Assured – Of Being Invaded: Web sites with and without privacy seals," in *Proceedings of the IADIS international conference e-society 2003*, pp. 63–71, June 2003.

[28] A. Kobsa and J. Schreck, "Privacy through Pseudonymity in User-Adaptive Systems.," *ACM Transactions on Internet Technology*, vol. 3, pp. 149–183, May 2003.

[29] J. Schreck, *Security and Privacy in User Modeling*. PhD thesis, University of Essen, 9 July 2001.

[30] M. De Boni and M. Prigmore, "Privacy and the Information Economy," in *Proceedings on IADIS International Conference on E-Society*, (Lisbon, Portugal), pp. 536–543, 2003.

[31] B. Schneier, *Secrets and lies: Digital Security in a Networked World*. John Wiley & Sons, 14 Aug. 2000.

[32] B. Schneier, "SSL Flaw," *Cryptogram*, Mar. 2003. http://www.counterpane.com/crypto-gram.html. Accessed May 12, 2003.

[33] A. Kobsa, "Personalized Hypermedia and International Privacy," *Communications of the ACM*, vol. 45, no. 5, pp. 64–67, 2002.

[34] C. Bettstetter, W. Kellerer, and J. Eberspächer, *Book of Visions 2000*, ch. Personal Profile Mobility for Ubiquitous service Usage, pp. 67–69. Wireless Strategic Initiative, 2000.

[35] B. Thai, R. Wan, A. Seneviratne, and T. Rakotoarivelo, "Integrated Personal Mobility Architecture: A Complete Personal Mobility Solution," *Mobile Networks and Applications*, vol. 8, pp. 27–36, Feb. 2003.

[36] I. Skender and D. Saric, "Provisioning and content adaptation of mobile data services," in *Proceedings of SoftCOM 2003, 11th International conference on Software, Telecommunications & Computer Networks*, (Split and Dubrovnik, Croatia, Venica and Ancona, Italy), pp. 596–600, IEEE, Oct. 2003.

[37] M. Koch and W. Wörndl, "Community Support and Identity Management," in *Proceedings of the Seventh European Conference on Computer Supported Cooperative Work*, (Bonn, Germany), pp. 319–338, Kluwer academic Publisher, Sept. 2001.

[38] Microsoft, ".NET Passport Review Guide." http://www.microsoft.com/net/services/passport/review_guide.asp, 7 Oct. 2002. Accessed April 12, 2003.

[39] Microsoft, "Microsoft .NET Passport for Businesses." http://www.microsoft.com/net/services/ passport/business.asp, 13 Mar. 2002. Accessed April 12, 2003.

[40] Novell, "digitalme." http://www.digitalme.com. Accessed January 12, 2004.

[41] A. Conry-Murray, "Microsoft's Passport to Controversy," *Network Magazine*, vol. 17, pp. 46–49, Mar. 2002.

[42] Liberty Alliance, "Liberty Architecture overview-v1.1." http://www.projectliberty.org/specs/archive/v1_1/liberty-architecture-overview-v1.1.pdf, 15 Jan. 2003. Accessed April 12, 2003.

[43] G. W. Bauer, "User data management." http://www.mozilla.org/projects/ui/ communicator/browser/wallet/, 16 Apr. 2003. Accessed March 27, 2003.

[44] R. Want, T. Pering, G. Danneels, M. Kumar, M. Sundar, and J. Light, "The Personal Server: Changing the Way We think About Ubiquitous Computing," in *Ubicomp2002: Ubiquitous Computing*, vol. 2498 of *LNCS*, (Göteborg, Sweden), pp. 194–209, Springer Verlag, 2002.

[45] XNSORG, "From Name Service to Identity Service: How XNS Builds on the DNS model." http://www.xns.org/urn-dns/tech-white-paper-urn-20020708.html, 9 July 2002. Whitepaper. Accessed January 13, 2004.

[46] Nokia, "Nokia 6130 phone features." http://www.nokia.com/cda10/ 0,4281,3305,00.html. Accessed January 26, 2004.

[47] M. Tjioe, "Symbian review: SmartProfiles." http://www.geek.com/hwswrev/psion/ smprof/, 26 Sept. 2002. Accessed January 19, 2004.

[48] D. Mulligan and A. Schwartz, "Your place or mine?: privacy concerns and solutions for server and client-side storage of personal information," in *Proceedings of the tenth conference on Computers, freedom and privacy: challenging the assumptions*, (Toronto, Canada), pp. 81 – 84, ACM Press, 2000.

[49] P. Jäppinen, "Bluetooth wireless technology based guidance system," Master's thesis, Lappeenranta University of Technology, Oct. 2001.

[50] T. Howes, M. Smith, and F. Dawson, "A MIME Content-Type for Directory Information." IETF Standard RFC 2425, Sept. 1998.

[51] F. Dawson and T. Howes, "vCard MIME Directory Profile." IETF Standard RFC 2426, Sept. 1998.

[52] SyncML initiative, "SyncML specifications 1.1." http://www.syncml.org/technology.html, 2002. Accessed January 13, 2004.

[53] U. Hansmann, R. Mettälä, A. Purakayastha, and P. Thompson, *SyncML: Synchronizing and Managing Your Mobile Data*. Prentice Hall PTR, Sept. 2002.

[54] D. Eastlake and T. Goldstein, "ECML v1.1: Field Specifications for E-Commerce." IETF Standard RFC 3106, Apr. 2001.

[55] Oasis Consortium, "Markup languages for Names and Addresses." http://xml.coverpages.org/ namesAndAddresses.html, 2 Jan. 2003. Accessed April 11, 2003.

[56] O. Berthold and M. Köhntop, "Identity Management Based On P3P," in *Designing Privacy Enhancing Technologies: International Workshop on Design Issues in Anonymity and Unobservability*, vol. 2009 of *Lecture Notes in Computer Science*, (Berkeley, CA, USA), pp. 141–160, Springer-Verlag, July 2000.

[57] E. van der Vlist, *XML Schema*. O'Reilly & Associates Inc, 2002.

[58] H. Kaaranen, A. Ahtiainen, L. Laitinen, S. Naghian, and V. Niemi, *UMTS Networks: Architecture, Mobility and Services.* John Wiley & Sons, Inc, 2001.

[59] Infrared Data Association, "IrDA standards." http://www.irda.org/standards/ standards.asp. Accessed August 4, 2003.

[60] D. Suvak, "Comparing the Benefits of IrDA and Bluetooth," *Wireless Systems Design*, vol. 5, pp. 31–36, May 2000.

[61] IEEE, "IEEE 802.11. hot topics." http://grouper.iee.org/groups/802/11/index.html. Accessed August 4, 2003.

[62] G. Held, *Data Over Wireless Networks: Bluetooth, WAP, and Wireless LANs.* McGraw-Hill Osborne Media, 2001.

[63] A. Dornan, *The Essential Guide to Wireless Communications Applications: From Cellular Systems to WAP and M-Commerce.* Prentice Hall PTR, 2001.

[64] S. McAteer, "In Defense of Bluetooth," *Incisor magazine*, pp. 9–10, Aug. 2003.

[65] J. Bray and C. Sturman, *Bluetooth: Connection Without Cables.* Prentice Hall PTR, 1 ed., 2001.

[66] Bluetooth SIG, "Bluetooth Specification 1.1." http://www.bluetooth.org, 2002. Accessed August 5, 2003.

[67] D. A. Gratton, *Bluetooth Profiles.* Prentice Hall PTR, 2003.

[68] S. Guthery, R. Kehr, J. Posegga, and H. Vogt, "GSM SIMs as Web Servers," in *Short-Proceedings of 7th International Conference on Intelligence in Services and Networks IS&N'2000*, (Athens, Greece), Feb. 2000.

[69] R. E. Smith, *From Passwords to Public Keys.* Addison-Wesley, 1 ed., 1 Oct. 2001.

[70] S. Garfinkel and G. Spafford, *Web Security, Privacy & Commerce.* O'Reilly & associates Inc, 2nd ed., 2002.

[71] R. Burton, *The Arabian Nights.* Modern Library, 25 Feb. 1997.

[72] Dallas Semiconductors, "iButton." http://www.ibutton.com/, 2003. Accessed January 2, 2004.

[73] Population registration center, "FINEID specifications." ttp://www.fineid.fi/default.asp?path=4%2CTechnical+information%2F8%2C Standards&file=0%2CFINEID%2Dspecifications%2Elink&template=, 2003. Accessed February 5, 2004.

[74] L. F. Cranor and B. A. LaMacchia, "Spam!," *Communications of the ACM*, vol. 41, pp. 74–83, Aug. 1998.

[75] Ericsson, "BlipNet : Technical overview." http://www.ericsson.com/about/blipnet/ technical_overview.pdf, Mar. 2003. Accessed July 17, 2003.

[76] B. Schneier, *Applied Cryptography.* John Wiley & Sons, 1995.

[77] S. Keski-Jaskari, P. Jäppinen, and J. Porras, "Applying Wireless Communication Technology to Fitness Devices," in *Proceedings of Workshop on Wireless Communications*, no. 158 in Acta Universitatis Lappeenrantaensis, (Lappeenranta, Finland), pp. 59–67, Aug. 2003.

[78] L. F. Cranor, *Web Privacy with P3P*. O'Reilly and Associates, 1 ed., 23 Sept. 2002.

[79] L. Edwards, "The Problem with Privacy: A Modest Proposal," in *Proceedings of the IADIS International Conference e-society*, vol. 2, (Lisbon, Portugal), pp. 699–704, 2003.

[80] J. Porras, P. Hiirsalmi, and A. Valtaoja, "Peer-to-peer Communication Approach for a Mobile Environment," in *Proceedings of the 37th Hawaii International Conference on System Sciences*, (Hawaii, USA), Jan. 2004.