UNIVERSITETET I AGDER

# A Framework for Identity and Privacy Management on Mobile Devices

Christian Hansen

Thesis submitted in Partial Fulfillment of the Requirements for the Degree Master of Technology in Information and Communication Technology

Faculty of Engineering and Science
University of Agder

Grimstad, Norway
August 2010

# Abstract

More and more online services require user identification. This increases time to fill out extensive forms and results in large amounts of login and identification data to remember. At the same time the number of users that need access to those service while roaming is equally increasing.

However, unfortunately many users are not aware that there is a high risk of loosing privacy when disclosing information about oneself's identity in an unregulated way. To counteract this and to help users in managing and maintaining related identity data, so-called *Identity Management Systems* have been developed. While available solutions are mainly built for fixed environments, dependencies to central storages and processing units make them unsuitable for application into mobile environments. Thus, a more flexible solution is necessary that supports roaming users with privacy-sensitive handling of identification processes in online transactions.

On this background, the project goal was an extension of the Identity Management System concept with mobility aspect. A framework for identity and privacy management on mobile devices, consisting of a *procedural method*, *privacy and security protocols* and a *user tool* has been specified to give users full control over their identity data in flexible and privacy-friendly ways. Thereby, the *method* has been defined to describe the overall process sequence. The supporting *protocols* then have been specified to provide ways for users and Service Providers to agree on applied data management practices, enable automated disclosures of identity data and guarantee secure and anonymous transmissions. Finally the *tool* has been defined to present an application to be installed on mobile phones that integrates the method and the protocols into a user-centered system architecture. Based on an engineering paradigm in combination with the first part of a six-step development strategy, this project covers the background research, requirements and specifications and design and development. This means that the final rollout of the proposed framework solution needs to be handed over to programmers in a possible project continuation. Those are then responsible for subsequent coding, testing and deployment.

After requirements and specifications had been derived, the framework has been successfully developed. While the user tool is responsible for all procedures on the mobile phone, a particular network infrastructure design allows secure transmissions by maintaining user anonymity. The solution is developed and the deployment prepared to such detail that programmers can directly start coding and testing.

As a conclusion, this project revealed several interesting and new aspects in the combined areas of identity, privacy and mobility. The solution fully meets all defined functional and non-functional requirements. As an application on mobile phones, the proposed framework allows privacy-sensitive handling of identity data in online transactions. Together with mechanisms for data management and maintenance before and after disclosure, it increases user flexibility, simplifies online identification and decreases processing time.

# Zusammenfassung

Durch die Zunahme von Onlineanwendungen mit Benutzeridentifikation steigt die Anzahl der Informationen, die für die verschiedensten Anmeldungen notwendig sind. Zudem erhöht sich die Bearbeitungszeit zum Ausfüllen der teilweise komplexen Anmeldeformulare. Im gleichen Atemzug steigt auch die Anzahl der mobilen Anwender, welche Anwendungszugriff benötigen.

Leider sind sich viele Benutzer nicht bewusst, dass die eigene Privatsphäre durch unkontrolliertes Preisgeben von persönlichen Informationen besonders online stark verletzt werden kann. Um in diesem Zusammenhang eine bestmögliche Unterstützung bieten zu können, wurden Identitätsmanagement-Systeme entworfen. Oftmals machen jedoch Abhängigkeiten zu zentralen Verwaltungskomponenten und Speichereinheiten die verfügbaren Lösungen untauglich für mobile Anwender. Es ist daher eine flexiblere Lösung gefordert, die einen hohen Schutz von Privatsphären im Umgang mit Onlineanwendungen ermöglicht.

Aus diesen Gründen war es das Projektziel, ein Identitätsmanagement-System mit erweiterter Mobilität zu entwickeln. Es wurde ein System entworfen, bestehend aus einer Vorgehensmethode, Sicherheitsprotokollen und einer Endbenutzeranwendung. Dabei wurde festgelegt, dass die Methode die grundlegende Vorgehensweise beschreibt und die Protokolle für Aushandlung von Datenverarbeitungsrichtlinien zwischen Anwendern und Serviceanbietern, automatische Freigabe von persönlichen Informationen und gesicherte und anonyme Datenübertragung verantwortlich seien. Für die Anwendung wurde definiert, dass sie die Methode und die Protokolle in einer Applikation für mobile Endgeräte zusammenführt. Grundsätzlich wurde das vorliegende Projekt an einer sechsstufigen Entwicklungsstrategie ausgerichtet, wobei der Schwerpunkt auf der Analyse, dem Festlegen von Anforderungen und Spezifikationen und dem anschließenden Entwickeln der Systemkomponenten lag. Im Umkehrschluss bedeutet dies, dass im Rahmen einer möglichen Projektweiterführung die entwickelte Lösung programmiert und getestet werden muss, um es schließlich einsetzen zu können.

Das gewünschte System wurde erfolgreich auf Grundlage der erarbeiteten Anforderungen und Spezifikationen entwickeln. Um Programmieren eine sofortige Bearbeitung zu ermöglichen, wurde das System in einem tiefen Detailierungsgrad konstruiert. Während die entworfene Anwendung für Prozessabhandlung auf dem Endgerät verantwortlich ist, ermöglicht eine speziell für dieses Projekt entwickelte Netzwerk-Infrastruktur sichere und anonyme Datenübertragungen.

Abschließend kann gesagt werden, dass dieses Projekt viele interessante und neue Aspekte durch die Kombination von unterschiedlichsten Untersuchungsfeldern aufgedeckt hat. Die vorgestellte Lösung erfüllt dabei alle funktionalen und nicht-funktionalen Projektanforderungen. Durch die Installation der Anwendung auf Mobiltelefonen erlaubt das vorgestellte System einen bewussten Umgang mit personenbezogenen, sensiblen Daten. Erweiterte Funktionalitäten ermöglichen zudem die Verwaltung von relevanten Daten auch vor und nach Offenlegung. Das vorgestellte System erhöht somit Endanwender-Flexibilität, vereinfacht Online-Identifikationen und verkürzt die Zeit beim Ausfüllen von Anmeldeformularen.

# Preface

This thesis was written as part of the master programme in Information and Communication Technology at the University of Agder, Faculty of Engineering and Science. It presents the results of my spring semester 2010 participation in the project course IKT590 with a workload of 30 ECTS credits. The idea behind this subject was mainly driven by my bachelor's thesis in that I developed a concept for cryptography and alternative authentication that allows protection of mobile devices. This time I wanted to apply the same platform with the aim to ease privacy-sensitive handling of personal data in online identifications.

This report was written under the supervision of professor Vladimir A. Oleshchuk and project manager Stein Bergsmark. Firstly, I would like to thank Vladimir for his useful technical input and the discussions about mobile security and verification. Vladimir's great security and privacy knowledge was very helpful during project execution. Secondly, I would also like to express my thanks to Stein for his intensive support that regularly improved my research and writing. Many sessions helped a lot to carry out this project in such a structured way and to achieve these satisfying results. Stein gave me very helpful insights into research work that I have not been aware of before I came to Norway. Thank you both again for your excellent supervision.

Grimstad, August 2010

Christian Hansen

# Contents

# List of Figures

# List of Tables

# CHAPTER 1

# Introduction

This work aims to support roaming users in online transactions that require identification. As a reason that in those situations very sensitive, personal information is disclosed, these activities require privacy protection on high levels. Thereby, the treated sub domain, representing the context of this project, lies in identification processes and privacy issues that need to be considered when using mobile devices.

The overall goal is to develop an effective and efficient framework for mobile, flexible Identity and Privacy Management functionalities. A procedural method, privacy and security protocols and a user tool simplify user-centered authentication in a privacy-friendly manner. In this way, the solution tries to mitigate disadvantages and challenges of traditional, centralized Identity Management System solutions when used by roaming users.

Chapter 1 introduces the project subject by presenting an overview of the working background in the first section.

The second section states the problem that needs to be faced during project execution. It clarifies thesis subject and goal, project objectives and the derived non-functional requirements. Furthermore, it discusses particular research questions that drive the entire work.

After this, the literature review is carried out. The presented information establishes the foundations for the background research in Chapter 2. It also provides a good insight into the treated project context.

Section four states the solution approach. It points out key assumptions and limitations and presents applied research and development methods. This section also clarifies the development strategy that is used during the entire project.

The contribution to the field of Identity and Privacy Management and the report outline conclude Chapter 1.

## 1.1 Background

Today's daily life is characterized by online services that require user identification. A steadily increasing number of these services leads to a simultaneous growth of the amount of login data that users need to remember and apply. Furthermore, the related registration processes in order to

create user accounts require considerable processing time. Last but not least, every single disclosed identity information discloses a part of the user's personal privacy at the same time.

Inspired by these facts, so-called *Identity Management System (IDMS)* have been introduced. They help users to manage and apply identity data in simplified ways. IDMS try to assure data confidentiality and to protect the user's privacy. Furthermore, they aim to reduce the volume of identity data that users need for accessing services. Remarkable systems in this context are probably [1], [2], [3], [4]. They, and many other systems, provide central storages that hold the users' privacy-sensitive identity data. Thus, they provide user support as long as users are connected to the corresponding resources. However, today's business is affected by people who spend most of their time away from fixed computers. Therefore, such centralized infrastructure approaches do not provide satisfactory solutions while roaming.

At the same time, almost everybody today owns at least one mobile device[1]. ITU Secretary-General Hamadoun Tour recently announced [5] that the number of worldwide mobile cellular subscribers was around 4 billion by the end of 2008. Moreover, it can generally be seen that more and more users tend not to use mobile phones only for making calls and writing messages. Inspired by enhanced mobility they have figured out several advantages in applying such devices to conducting online transactions and accessing resources.

So, why not using the mobility aspect of such personalized devices by applying them as trusted control interfaces for services that require privacy-sensitive identity data? Going one step further, why not even seeing these devices as the new generation of more flexible IDMS solutions?

Based on all the above facts, the following project statement is defined.

> The project context lies in identification scenarios that are carried out by roaming users on mobile devices. Thus, it treats a sub domain of the large area working with traditional IDMS infrastructures that support comparable processes in fixed environments. The intended solution aims to combine the key advantages of IDMS technology with the enhanced flexibility of mobile phones.
>
> An effective and efficient framework is aspired that equips these devices with corresponding identity and privacy related mechanisms. Thereby, the framework is planned to consist of a *procedural method* to specify the overall process sequence, *privacy and security protocols* to mainly protect transactions and data, and a *user tool* application for mobile phones to combine these two framework parts and extend them with an appropriate user interface. The solution needs to ensure roaming users full control over their identity data in flexible and privacy-friendly ways. Moreover, mobility reasons also require that all identity data is securely stored directly on the mobile devices.

## 1.2 Problem Statement

This section presents the project problem statement. It defines the thesis subject with its main goal, as well as a set of supporting objectives. Furthermore, non-functional requirements for the

[1] Following, this report uses the terms *mobile device* and *mobile phone* equivalently.

development process are specified - the functional requirements are later on worked out during the requirements analysis (Chapter 3). Finally, four research questions are posed that define the direction of the work.

### 1.2.1 Thesis Subject and Goal

The traditional IDMS approaches which are mainly used today can not be efficiently and simply applied to mobile users. The need for permanent connections to central storages and processing units makes them unusable or at least very challenging while roaming. Because of this, a more flexible solution is required. Therefore, the technology platforms of mobile devices are used as trusted control interfaces. Applied as identity proxies and agents they support secure data exchanges while preserving high privacy levels. Furthermore, they enable protected storages for privacy-sensitive identity data that is integrated into mobile phones. A user-centered approach guarantees self-responsible data management and handling, and an easy and transparent user tool allows different degrees of privacy, anonymity, accountability and confidentiality. This information leads to the following main project goal that aims to mitigate the above shortcomings.

**Development of a framework for Identity Management (IM) and Privacy Management (PM) on mobile devices. The solution provides an overall procedural method, privacy and security protocols and a user tool for mobile phones. In addition to this, ways to securely store and anonymously exchange identity attributes are specified and designed.**

This project covers the first parts of a six-step development strategy (see Subsection 1.4.5. Even though it thus presents the entire design of a theoretical framework for IM and PM on mobile devices, there are still some further steps necessary in order to finally complete and deploy the solution. That means, that after project closure, the proposed design needs to be coded and tested by programmers. As soon as the designed subsystems of Chapter 3 are integrated into the underlying infrastructure, the user tool is ready to be rolled out. As an installed application on mobile phones it then allows privacy-sensitive handling of personal identity data in online scenarios. A detailed discussion about project coverage and subsequent development tasks is shown in Subsection 1.4.5.

As a conclusion, this goal mentions three framework parts. In order to provide a quick insight into their meanings, the following list clarifies related functionalities.

**Procedural Method** - Describes the process sequence to use mobile devices as trusted control interfaces for online transactions that require privacy-sensitive identification.

**Privacy and Security Protocols** - Support the procedural method. Provide a way for users and Service Provider (SP)s to agree on data management practices. Enable automated disclosure of identity attributes and guarantee secure and anonymous data exchanges of privacy-sensitive identities.

**User Tool** - Comprises the central framework part. Integrates procedural method and privacy and security protocols into a user-centered system architecture. Provides interfaces and mechanisms to manage and handle identity attributes that are stored on the mobile devices, according to various privacy requirements.

### 1.2.2 Project Objectives

To realize the overall goal of developing a framework for IM and PM on mobile devices, the following set of related project objectives is defined.

**Background Research**

- Analysis of the working area, resulting in a list of main advantages, disadvantages and challenges in the use of mobile devices for IM and PM purposes.

**Requirements and Specifications**

- Definition of a set of requirements for the procedural method, privacy and security protocols and user tool that is applied to the framework development.
- An analysis of existing solutions for the framework parts against the overall project goal and the non-functional requirements in order to identify potential ways of adaptation.
- Definition of a set of requirements for underlying infrastructure and applied technology that allows to deploy the framework into a larger context.
- Statement of a set of requirements for SPs that enables them to adapt their services to be used in the framework solution.

**Design and Development**

- Adaptation of existing or development of new solutions for the three framework parts.
- Determination of a way to safely store privacy-sensitive data directly on mobile devices.
- Integration of the framework parts to build up the overall solution.
- Identification of the extent to which mobile devices can be used for IM and PM purposes.

**If time allows: Coding**

- Proof of concept development of a prototype according to the proposed framework design.

### 1.2.3 Non-Functional Requirements

The following list represents six non-functional requirements for the project solution.

NFR 1. It is executable on today's mobile phone platforms and technologies and does not require significant changes in the operating system infrastructure[2].

NFR 2. It supports different degrees of privacy, anonymity, accountability and confidentiality.

NFR 3. It applies protocols that are, whenever possible and practical, based on standards.

NFR 4. It proposes a user tool that is easy to understand and simple to use with basic knowledge about standard functionalities on mobile phones (e.g. starting programs or surfing the web).

NFR 5. It consumes as little power as possible.

NFR 6. It specifies requirements for SPs based on minimum complexity and cost that enable them to adapt their services to the framework solution.

[2]Based on time restrictions only one particular mobile phone platform is treated.

### 1.2.4 Research Questions

According to the main project goal (Section 1.2), the derived objectives (Subsection 1.2.2) and the non-functional requirements (Subsection 1.2.3), four research questions are stated. Successfully answering them demonstrates that the proposed solution provides an effective and efficient framework for IM and PM on mobile devices.

1. **What are the main advantages, disadvantages and challenges when using IM and PM on mobile devices?**

   An effective and efficient framework is designed that helps to meet the challenges of IM and PM for roaming users. Thus, specifications of general advantages, disadvantages and challenges and in particular, mobile related ones build up the basis for the entire development process.

2. **What are the requirements for the framework parts - the procedural method, privacy and security protocols and user tool? And do corresponding solutions exist that can be adapted or is it necessary to develop them?**

   The framework consists of a procedural method, privacy and security protocols and a user tool. Here, existing solutions are analyzed, to define development requirements. Possible problems in the use of mobile devices for IM and PM that have already been researched are also taken into account. Furthermore, it is identified if existing methods, protocols and tools can be adjusted and adopted. If there is none suitable solution available the framework parts are carefully designed and developed so that they fulfill the demands of roaming users. Finally, all parts are required to work closely together in order to achieve a coherent overall solution.

3. **What kind of communication infrastructure and technology is needed and what are the requirements for SPs to enable application of the framework solution?**

   In order to deploy the proposed framework into a larger context, the necessary underlying communication infrastructure and applied technology are specified. In addition to this, external requirements are derived from the framework solution so that SPs are able to modify their services and efficiently support roaming users.

4. **To what extent does the framework solution enable roaming users with proof of identity while preserving high privacy levels?**

   Using mobile devices for IM and PM purposes and as secure storages for privacy-sensitive identity data allows users flexible IDMS solutions while roaming. Therefore, the field of application for the proposed framework is analyzed.

## 1.3 Literature Review

The IM and PM research areas are mainly relevant for this project. This section gives a short overview of important work published in those areas. It helps to establish the Background Research (see Chapter 2) that builds up the scientific basis for the entire project. It also provides an impression of the treated project context in the large field dealing with IM and PM. While in the Background (Section 1.1) general IDMSs have been addressed this section focuses on solutions especially designed for mobile devices. As it is seen later, IM and PM aspects are applied as combined and related features in this project. However, distinction at this point leads to a better overview of actual research.

### 1.3.1 Identity Management

An IDMS prototype for Personal Digital Assistant (PDA)s was developed at the University of Freiburg [6]. It allows to create different so-called *partial identities*[3]. Whenever a service requests identification, the tool interactively selects a suitable identity from the pool of preconfigured ones. If none applicable partial identity exists, it allows users to create a new one. Furthermore, the system warns when unnecessary identity attributes are disclosed[4]. However, so far this prototype is mainly based on preconfigured, fixed identities that only support limited services.

In his paper [7] Hyppönen proposed a mobile IDMS that stores privacy-sensitive data in Subscriber Identity Module (SIM) cards. The tool was created with particular focus on the privacy design requirements *data minimization* and *informed user consent*. It is called *open*, because it can be "joined and freely used by any participating party". Even if it may be open to join it unfortunately has the main disadvantage that it requires modified SIM cards. While they provide high data security they come along with increased development costs.

Paci et al. developed a mobile IDMS with main focus on privacy preservation of digital identities [8]. The key concept consists of a SP application, a client application and a registrar. In contrast to the two previous solutions identity attributes are not stored on mobile devices but rather hosted by the central registrar. To establish connections, the client application contacts SPs who in turn request identity proofs from the registrar. Only if that is successful, connections can be established. In this approach the registrar is the weakest unit that hosts privacy-sensitive data of all participants. Users are required to entrust all their identity attributes to it.

The last presented research work is primary built for Near Field Communication (NFC)[5] [9]. Like the solution listed second, this one also uses SIM cards to store sensitive information. But in contrast, it works with long lived public key certificates as identities. Each user is allowed to own only one pseudonym for all identification processes. The main disadvantage is that the certificates are not dynamic. Thus, after initial issue, no information can be updated, deleted or added. Furthermore, revocation of incorrect information has not been implemented satisfactory so far.

Those four representations conclude the overview of the IM related research field. Two further solutions that were left out for reasons of limited space are [10] and [11]. Summing those works up, all have one or more disadvantages that prevent efficient use on mobile devices. Generally, these are:

- Limitation to a small amount of preconfigured services.
- Limitation to preconfigured and fixed pseudonyms or application of static certificates.
- Requirement for user trust into central registrar.
- Expensive developments caused by modified, personalized SIM cards.

[3] *Partial identity* is a refinement of the term *digital identity*. It consists of a combination of one or more identity attributes for a particular context.

[4] This functionality is based on the Platform for Privacy Preferences Project (P3P). However, privacy related aspects are out of the current discussion and rather analyzed in the next subsection.

[5] NFC enables communication of two objects that are placed closely together. For more information it is referred to `http://www.nfc-forum.org`.

### 1.3.2 Privacy Management

Berthold and Köhntopp [12] proposed to develop a privacy aware IDMS based on P3P[6]. They suggested extensions of the standard P3P vocabulary to cover mobility aspects. Moreover, they defined to use digital pseudonyms that are certified by central Certification Authority (CA)s. For high security and privacy they recommended to apply a chain of multiple CAs in that each CA certifies another one. This work only presents theoretical ideas that were neither tested nor verified. However, some important aspects for this project can still be extracted.

The position paper [14] written by Bandara et al. describes the *Privacy Rights Management for Mobile Applications (PRiMMA)* project[7]. The framework in development allows users to manage protection levels of privacy-sensitive information generated by pervasive systems [15]. Like Digital Rights Management (DRM) the proposed system integrates policies into privacy-sensitive data to make information flows fully controllable. At point of writing this project is still in progress. Even though so far some important implementation aspects are still outstanding, useful input for privacy requirement specifications within this project can be derived.

Dynamic PM in form of a plug in service for the middleware in pervasive computing was described by Hong, Y. And Shen in [16]. Like [12], different privacy levels are achieved by extending the standard P3P vocabulary. It's main focus is to avoid privacy issues related to context and location sensitive information. Thereby, the middleware is responsible to compare P3P policies of requested services with user preferences. And a user interface that needs to be installed on personal computers enables to create those preferences that represent different privacy levels. This system provides very valuable input for PM related aspects in this project. As shown later, the idea of using P3P is entirely adopted.

The last, and by far most difficult PM architecture to be presented, works below the transport layer of the Internet Protocol (IP) stack [17]. Based on the *Open Systems Interconnection (OSI) reference model*[8] it works with pseudo random identifiers in network layers. Protocols above this layer that are responsible for data exchanges are encapsulated and enciphered with a further protocol. The authors designed two different modes to support enhanced privacy and to enable context dependent privacy. While the "normal" mode only provides privacy protection through applied protocols, the "stealth" mode also creates a random Media Access Control (MAC) address and a temporary IP address. Anyway, this approach is not that helpful for the intended framework. It is mainly mentioned, to show an entirely different privacy approach.

These PM related systems emphasize on a variety of potential solutions. In contrast to the IM infrastructures there are no disadvantages to point out. The works rather show important aspects that are later on taken into account when designing privacy related framework functionalities.

[6] P3P is a standard to agree on data management practices that are applied to exchanges of personal data. It is developed by the World Wide Web Consortium (W3C) [13]. P3P is discussed in more detail in Subsection 2.2.5.

[7] The project webpage is located at `http://primma.open.ac.uk/`.

[8] The OSI reference model specification is published at `http://standards.iso.org/ittf/PubliclyAvailableStandards/s020269_ISO_IEC_7498-1_1994(E).zip`.

## 1.4 Solution Approach

In this section the solution approach is shown. It specifies the key assumptions and the limitations for project coverage. The section then states applied research and development methods. Finally, the development strategy is clarified that is used throughout the entire project. All information helps to make this work as transparent as possible. This enables easy handover of the theoretical framework solution to programmers in order to allow coding, testing and deployment.

### 1.4.1 Key Assumptions

These three key assumptions provide the foundation for the work:

- The proposed framework works within best privacy levels possible. However, *complete* privacy can not be guaranteed[9].
- The used mobile operating system is required to enable installations and executions of third party software.
- Dependent on the chosen solution adjustments on SP side need to be realizable. In this context, the necessary prerequisites to enable SPs with framework adaptation are specified during project execution, but it is out of the scope to carry out any of these changes.

### 1.4.2 Limitations

The following aspects limit the field of study and define the sub domain of IM and PM for this work:

- The project develops a theoretical framework for IM and PM on mobile devices. The practical counterpart by means of coding is excluded (exception: prototyping if applicable).
- While considering security and privacy issues, usability is also consulted but less, if any weight is put on this during design and development.
- Location privacy is considered but not discussed and processed in detail as a self contained issue[10].
- In case third party software is used, it needs to be open source.
- Scalability issues are excluded from the entire development.
- Possible discrepancies between using mobile devices for private and business purposes are out of the project scope.

[9] The extent of *complete* privacy is very hard to define. It is strongly dependent on the current context and the affected user. Furthermore, it needs to be individually identified who the *enemies* could be and to what degree related data needs to be kept private [18]. In the same context, mobile phones unfortunately have the potential to threat user privacy [19].

[10] Even though location privacy is important when dealing in the mobile area, it is not of that importance for this project. This is, because the intended solution mainly works in a different context.

### 1.4.3 Research Method

Before any process in this project can successfully be started, an insight into structured working is required. This is to say, how to carry out research[11]. Though, commonly used *qualitative* and *quantitative*[12] research approaches have proven to be effective. With the main project focus on theoretical considerations rather than coding, the qualitative method is considered to be most appropriate. It is thus applied in data collections for the Literature Review (Section 1.3) and the Background Research (Chapter 2). Furthermore, research questions were defined in Subsection 1.2.4 rather than hypotheses that are used in the quantitative approach.

In this project the used process sequence is therefore generally derived from qualitative research. But whenever applicable methods from the quantitative approach are supplemented. The applied sequence consists of three main steps, as shown following. However, they are assumed to be known to readers and thus not described in more detail in this report. It is rather continued with facing subsequent tasks within design and development.

1. Collect data
2. Analyze data
3. Present recommendations and improvements

### 1.4.4 Development Method

The basic development method used in this project is a four-step engineering paradigm. It is probably one of the most often applied approaches when designing and developing new systems in the information technology. Usually, and this is also the case here, the following steps are performed in several loops, as described in the next paragraphs.

1. State requirements
2. State specifications
3. Design, document and implement the system
4. Test and validate the system

#### Engineering Paradigm

In step one functional and non-functional requirements are defined. Functional requirements represent high level descriptions of project modules that have to be included to make the system functional. They specify how the system should behave. Non-functional requirements on the other hand are general quality and performance characteristics for the solution. An example is *low power consumption*, as defined in Subsection 1.2.3.

The second step then derives specifications from those requirements. While requirements are mostly user oriented, specifications focus on the technical implementation. They define what resources will be used to meet and implement the stated requirements. Specifications thus describe underlying blocks and mechanisms that are needed to physically realize the framework.

[11] The aim of research is to "gather information to answer a question that solves a problem" [20].
[12] It is assumed that the reader is aware of those techniques. Since, no detailed information is given in this report.

Once requirements and specifications are stated, the design process is carried out. For that, this project applies a *top down* development. This means, that it starts with presenting an overall procedural method. It then designs privacy and security protocols that are required to enable the presented method. Finally, it creates a user tool that combines method and protocols and adds interfaces and mechanisms.

The last engineering step is not completely covered by the project scope, but validity is still discussed in Chapters 4 and 5. In addition, the entire development strictly follows two standardized design guidelines, as presented in Subsection 2.2.4. By conforming to them it is tried to carry out every single process step correctly. In this way the probability is increased that the presented solution is valid, even though no common verification approach is applied. In any case, practical validation and verification can only be done, when subsequent coding of the first prototypes has been completed.

#### Security Related Methods

With special focus on verification purposes, the project subject requires a look into the security area. Thereby, it was analyzed that there is indeed particular research going on within security verification (e.g. [21] and [22]). But these studies are very complex since security systems are not only developed with respect to specific requirements. They rather also require appropriate possibilities to claim that proposed solutions are in a way secure by design. But formulating and satisfying security requirements in a provable way is very complicated.

Frequently applied methods in this area are security analyses, in most cases in informal ways. This can also be very challenging when arguing that a proposed solution is actually secure. However, going into that direction is out of the scope of this work. And this leads to the result that every security and privacy related design process (in this project especially protocols) is carried out and validated according to the aforementioned methods.

### 1.4.5 Development Strategy

In order to place the project into an overall development cycle, Figure 1.1 visualizes the applied strategy. The steps in yellow color represent tasks that are covered by this project[13]. The blue colored steps are the ones that follow after framework handover to programmers.

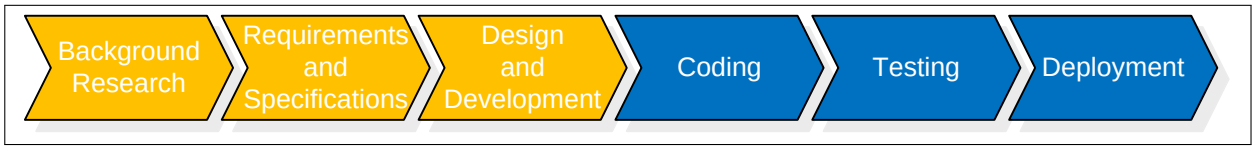



Figure 1.1: Development strategy

[13]The descriptions are also used as report headings. This helps to easily determine current project steps.

### Background Research

An extensive background research establishes basic project knowledge. Prior work that has already been carried out in the two operational areas within IM and PM is analyzed. Besides this, definitions, guidelines and technology approaches are identified. Achieved information gives a good overview of the current research status. It builds up the scientific basis for this work. The data also helps to define requirements and specifications in the next development step. However, it is important that research is not limited to system approaches for the mobile sector. Indeed, also traditional, centralized and fixed solutions are used. This allows to identify potential advantages that can be adopted and disadvantages that have to be avoided and rather be improved. Research results then show whether adaptation of existing solutions is useful or if new development is required. However, decision criteria for this are mainly specified in the next development step. So, the need for iteration, as stated in the previous subsections, becomes clear.

### Requirements and Specifications

Concerning requirements and specifications the most important aspects are already discussed in the engineering paradigm of Subsection 1.4.4. In addition, according to the previous development step it is now clarified where the main input for them is derived from. The results of the requirement and specification provide the baseline for the framework design and development. They enable comparisons of different solutions and represent demands for the overall system solution.

### Design and Development

How design and development is done according to requirements and specifications is also clarified in Subsection 1.4.4. The goal during this project is to design the overall system and all required subsystems so fine grained and specific that handover of the project solution to programmers is made as simple as possible. Thus, to smooth this transition and to guarantee successful deployments, Unified Modeling Language (UML) diagrams, process sequences, interfaces, various figures and detailed descriptions of all included mechanisms are provided. Besides this, requirements for SPs are worked out to enable services adaptation to the proposed framework. Furthermore, necessary specifications for underlying communication infrastructure and applied technology are stated.

### Subsequent Steps: Coding, Testing and Deployment

Resulting from limited time, this work concludes with the framework design and development. Thus, the last three development steps are outside the project scope. However, with the overarching goal of a system roll out, all theoretical framework designs are completed in that way that they easily can be handed over to programmers. Those are then responsible to code the proposed privacy and security protocols and the user tool according to the provided designs.

As soon as first prototypes are created, they need to be extensively tested. Even though validation of the design is already covered to the widest extent by this project, some improvements may only become visible when prototypes are available. Furthermore, it is not possible to theoretically verify every single framework aspect in sufficient detail.

It is most likely that coding and testing will require several iterations. But after all results are satisfactory and all requirements and specifications met, the framework solution can finally be rolled out. In this development step, programmers together with system architects are responsible for implementation of specified infrastructure units and to install the user tool on mobile phones. It is also their job to provide SPs with any material that is necessary to adjust their services accordingly. Various test runs finally conclude the deployment and also the development strategy of Figure 1.1. In Section 4.4 all briefly addressed steps are reviewed. And based on the framework design of Chapter 3, concrete tasks and requirements for the participants are added.

## 1.5 Contribution

Because of the steadily growing demand for IM and PM functionalities for mobile users, extensive research in this area is necessary. It is foreseeable, that mobile IDMS solutions finally will impact daily lives for many users. This assumption is based on the general perception that mobile devices can now be seen as personal belongings that follow users everywhere. Quoting Neil A. McEvoy [23]: "The single most important property of the mobile phone is that everybody already has one and they do not leave home without it." In addition, online services that require identification of users are strongly increasing, too.

Based on these facts, a combination of IM, PM and mobile devices can provide user friendly, privacy preserving and flexible solutions for roaming users. However, the majority of traditional and today's most frequently used IDMSs is built for online services and fixed environments. And those solutions are unsuitable while roaming, because they require permanent access to particular resources.

To help getting further in this interesting and important research field, the presented work extends mobile devices with IM and PM functionalities. then the user experience will be an application which allows browsing the internet in the usual way. However, different interfaces and the underlying system architecture guarantee easy, privacy-sensitive and secure disclosures of personal information. On this background, Subsection 1.3.1 presented the main disadvantages of other solutions. Mitigating those and supplementing additional benefits comprise the contribution of this project, as shown following.

**Storage** - In contrast to traditional IDMS solutions and the ones shown in the aforementioned subsection, the proposed framework stores privacy-sensitive identity data directly on mobile phones. This provides high flexibility and great impact on enhanced user trust.

**Pseudonyms** - An entirely different overall approach is proposed here. Instead of using fixed pseudonyms that are matched against requests, the designed solution enables application of individual pseudonyms for every single transaction. This guarantees that always only minimum amounts of data are disclosed.

**Risk Analysis** - The framework integrates mechanisms that identify if SPs have already received identity attributes. This new functionality, that is not embedded in any of the other solutions, helps in the analyses of potential privacy risks and issues issues that result from past transactions. In this way, it allows to work on high privacy levels.

**User Trust** - The solution works with the lowest amount of user trust relations as possible. Especially when applying privacy-sensitive data this is a very important aspect in order to achieve wide customer satisfaction. However, none of the other works emphasized this aspect so far.

**User Application** - A user-centered application for mobile phones allows users to fully control their identity data. It is important that this it is just as same essential to retain control during the entire data life cycle and not only before and during transactions. This includes a novel option for managing and maintaining disclosed identity data on SP sides.

**Operational Field** - While the presented mobile systems are only applicable to a few pre-configured services, the framework aims to limit the operational field as little as possible. Granted that SPs agree to minimally adjust their services and infrastructures, all online transactions that can be establish by browsing websites are supposed to be covered.

As a conclusion, a framework that integrates all these contributions is required. When users follow the solution's specifications, great advantages can be achieved. They benefit from proven traditional IDMS functionalities without needs of permanent access to centralized storages and processing units.

## 1.6 Report Outline

The rest of this report is structured as follows.

**Chapter 2 - Background Research**
The second chapter is divided into the two main study fields. Discussion of *Identity Management* and *Privacy Management* provide the scientific and theoretical basis for this work. The chapter shows advantages, disadvantages and challenges of traditional systems and also aspects that are particularly important when applying those techniques in the mobile area. By describing terms and definitions, design fundamentals and design goals for the framework, Chapter 2 gives an idea of how the solution can be built and behave. A couple of privacy related design requirements are then specified that lead to the framework development, shown in the next chapter

**Chapter 3 - Framework Development**
Chapter 3 presents the solution in terms of framework design and development. It shows the integration of IM and PM functionalities into mobile devices. Typical user scenarios are applied to visualize project related problems, challenges and goals and to emphasize on user benefits. This chapter also specifies the functional requirements for the three framework parts and necessary underlying infrastructure and applied technology. Then it presents different implementation ways for the framework solution. By including reference systems, contributions and benefits of the proposed system are clarified. A framework summary concludes this chapter. It enables smooth handover of design and specifications to programmers.

**Chapter 4 - Results**
The forth chapter presents the results of the proposed framework solution. It reflects the background research, the requirements and specifications analysis and the design and development step. In this way, the chapter shows how the solution fulfills requirements and specifications.. It also discusses the handover of the framework design to programmers in order to code prototypes, test and finally deploy the proposed system solution.

**Chapter 5 - Discussion**
Chapter five discusses the results of Chapter 4, starting by deriving discussion criteria based on the previous chapters. The criteria are then first matched against current systems in the working area and second against the framework solution. A comparison of both results emphasizes on the framework's improvements to other approaches and it's potential disadvantages. This chapter also shows the project limitations and it's advantages by means of user benefits. Furthermore, it describes extensions to the proposed system that will be part of future development.

**Chapter 6 - Conclusion**
The sixth chapter gives the conclusion of the study. It reminds of the motivation and the project problem and relates this to achieved results. The chapter shows project conclusions and contributions to state important implications for users and SPs. It also identifies advantages for SPs that decide to participate in the framework solution. An application roadmap then reviews the project handover to programmers by demonstrating subsequent tasks and responsibilities on the way to the framework roll out. Thoughts about planned improvements and recommendations for further research in the project field complete this chapter and the entire work.

# CHAPTER 2

# Background Research

Chapter 2 presents the theoretical background for this work. It establishes fundamental knowledge that is necessary to follow the upcoming framework development. Thereby, given information is based on the introduction of Chapter 1 and especially the literature review that is presented in Section 1.3.

The two main sections of this chapter discuss the project modules *Identity Management* and *Privacy Management*. They follow a similar layout structure and both start with describing related terms and definitions. This is followed by an analysis of requirements, design fundamentals and goals to visualize the directions of the proposed framework. Next, each section emphasizes on important system architecture approaches to clarify the project context. While the identity related section lists important advantages, disadvantages and challenges for using Identity Management and Identity Management Systems, the privacy related one identifies essential privacy issues and defines important Privacy Management Framework Design Requirements. A brief review concludes each one of these sections.

A conclusion ends this background research. It reviews the development strategy of Subsection 1.4.5 and thus points out the current project progress status and following tasks.

## 2.1 Identity Management

This section presents the basics within the working area of identity. It starts with an introduction of the most relevant terms and definitions. A description of participating parties in related infrastructures follows. The section also emphasizes on general accepted design goals for systems dealing with identity data. A short insight into two former architecture approaches rounds up the first part of Section 2.1. All information presented until this point can be seen in addition to the Literature Review given in Section 1.3.

The second part of this section starts with a very important discussion about *general* advantages, disadvantages and challenges of systems that help to manage identities. It then analyzes the same aspects with focus on *roaming users*. The overall aim of this is to drive the framework development and to answer the first research questions.

The conclusion then gives a brief outlook into other research fields working with the management of identities. The information once more shall visualize the treated project context.

### 2.1.1 Terms and Definitions

This subsection briefly shows the terminology within IM that is important for the carried out research. It discusses definitions of the terms *Digital Identity*, *Identity Management* and *Identity Management System*. Each description consists of two parts; a gray colored box and a running text. The aim of the box is to shortly summarize the essential facts of each term. The running text then presents the most relevant background for those facts.

#### Digital Identity

**Digital Identity** *Consists of privacy-sensitive identity attributes. Represents an individual person. Can be used for identification and authentication purposes. Needs to be protected on high security and privacy levels.*

Users daily request wide ranges of different services at various SPs. They also require access to lots of systems and resources. To register new accounts or to carry out transactions users have to authenticate against SPs with privacy-sensitive, personal information. This personal information can be defined as *identity attributes*. As soon as one or more identity attributes are disclosed, the overall term *identity* is used. Moreover, working within information technology it is most common to describe this as a *digital identity*. Users are allowed to have one or more digital identities in different contexts. Because of their sensitive content it is important to protect those digital identities on high security and privacy levels. Identity theft and misuse[1] have to be avoided or at least made difficult. Furthermore, it is essential that users consciously handle digital identities in order to circumvent privacy risks when disclosing information. However, privacy related discussion is separately carried out in Section 2.2.

#### Identity Management

**Identity Management** *Defines specifications for effective, efficient and secure handling of privacy-sensitive digital identities.*

It can generally be seen that almost every user who holds a variety of different digital identities seems to be overwhelmed to appropriately work with this. It often results in feeling bothered by the overload caused in memorizing and applying large amounts of account data. To describe the practical user support in this situation, the term *IM* was introduced. IM thus helps users to manage and maintain digital identities. It provides a set of definitions for identity related processes, services and technologies. Thereby, the most important idea is to support the entire digital identity life cycle that includes creation, maintenance and termination. IM is also responsible to specify ways to protect access to privacy-sensitive data and to secure data exchanges on high security levels.

[1] *Identity theft* and *identity misuse* are terms to describe frauds pretending to be another person. This way someone tries to get access on the basis of a different personal identity.

#### Identity Management System

**Identity Management System** *Provides the technology to integrate specifications that are defined by the Identity Management. Helps users in applying different digital identities, dependent on their current context.*

*IDMS* is the term for the technology that integrates all IM specifications. It establishes environments and rules to handle digital identities and their entire life cycle. An IDMS helps users to apply suitable identities based on varying situations and contexts. It also provides and applies secure protocols to protect data exchanges. Last but not least, it is common to see the overall goal of an IDMS in fulfilling the requirements defined by the Informational Self-Determination[2].

### 2.1.2 Participating Parties and Design Goals

Now that the three most important terms and definitions are clarified, this subsection introduces parties that comprise IDMS infrastructures. A description and visualization of a sample service request presents related tasks and process steps. This is followed by a specification of commonly used design goals for identity related systems developments. In this project defined as the *Four Basic Framework Pillars within IM* they have great impact on the entire solution.

#### Participating Parties

To successfully provide service, an IDMS needs to involve different parties, whereby each one is responsible for particular tasks. The following list is based on [24]. It briefly shows three very typical IDMS participants. However, the existence of the identity provider is dependent on the chosen solution and not always necessary.

**User** - Tries to get access to a system or a resource or requests a service.

**Identity Provider** - Issues digital identities; but this can also be done by users themselves.

**SP** - Provides identity checks and proofs and responses to service requests.

Based on those parties, the following steps (adapted from [25]) are typical within IDMS infrastructures, when users request access or service.

1. *User* makes initial registration with *Identity Provider* (this step is only carried out once; it can be skipped in following requests).
2. *User* sends request including personal identification information (digital identity) to *SP*, asking for access or service.
3. *SP* requests *Identity Provider* for validation of *User's* digital identity.

[2]The term *Informational Self-Determination* was published by the German Federal Constitutional Court. It deals with the collection of personal information and can be seen as the "right to privacy". The term is just mentioned here to clarify project connections; a separate discussion is rather placed in Subsection 2.2.4.

4. *Identity Provider* analyzes whether digital identity recently has been validated or not. If validation has not been carried out or validation time stamp is not updated enough, this is caught up (by identity data exchange between *Identity Provider* and *User*).
5. *Identity Provider* replies SP with results of verification check.
6. If appropriate *User* is not registered with requested *Identity Provider*, *Identity Provider* forwards request to another *Identity Provider* (this requires appropriate infrastructures).
7. Dependent on validation check, *SP* provides *User* with requested access or service.

Figure 2.1 demonstrates this process sequence when requesting access or service. It also visualizes the roles of the three addressed parties.

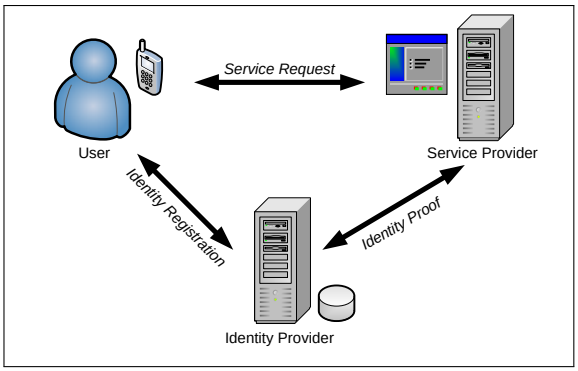



Figure 2.1: Participating parties and process flows in IDMS infrastructures

### Design Goals

Every well structured development process is based on design rules and goals. [26] presents commonly agreed goals for IDMS infrastructure and technology developments. In this project they are slightly adjusted and defined as the *Four Basis Framework Pillars within IM*. As Figure 2.2, it is necessary to keep the balance between all pillars. If this is the case, the solution is on a good way to be coherent, stable and consistent.

**Security** - Users need to be allowed to select security levels dependent on services and contexts. Critical services have to require and provide higher levels non-critical ones. For efficiency reasons, support and application of security levels needs to require minimum changes (if any) in underlying infrastructure and applied technology.

**User Trust** - When dealing with privacy-sensitive data, high user trust levels are essential. Even though those trust relations are required in all directions amongst users, SPs and Identity Providers they need to be limited whenever possible.

**Cost Efficiency** - IDMS solutions need to be cost efficient, by primary means of deployment and maintenance. But it is well known that high security almost always requires high investments. Finding the balance between cost, security and added value is therefore critical for broad acceptance and wide success.

**Ease of Use** - Simple deployment, maintenance and overall system architectures are as important as cost efficiency to reach satisfaction. The entire solution and especially process steps need to be transparent to all involved parties at all times. Users need to be able to realize how to act appropriately in particular situations. This means, that the system needs to be natural to operate and easy to use.

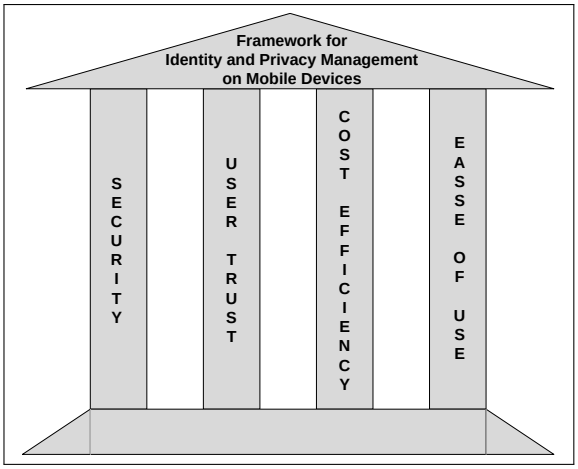



Figure 2.2: The four basic framework pillars within IM

### 2.1.3 System Architecture Approaches

To get an insight into the evolution and to show the growing demand for IM techniques, this subsection presents two former system architecture approaches. They can be seen as starting point for all research within IM. In combination with the literature review of Section 1.3 they also present fundamental drivers and general requirements for this project. Furthermore, they again help to place the work into bigger context, by means of the large domain dealing with IM.

A brief description of two different implementation strategies follows; centralized and decentralized architectures. Related advantages and disadvantages show the direction of the framework development and conclude this subsection.

#### Former Approaches

Already in 1985 researchers wondered about ways to reduce disclosure of privacy-sensitive data. However, they did not mention terms like IM or IDMS (or even PM[3]). Back then Chaum argued [27]: "The large-scale automated transaction systems of the near future can be designed to protect the privacy and maintain the security of both individuals and organizations." He suggested to use a different digital pseudonym with every communication party. The aim of those pseudonyms was to provide a way to control exchanges of privacy related data. Chaum stated that in contrast to other solutions known to him at that time (e.g. pseudonyms issued by organizations) he wanted to

[3]Privacy related discussion follows in the next section.

use self issued pseudonyms. He therefore defined specifications for a new computing device. This device, comparable to a PDA known nowadays, was meant to provide users with functionalities to create pseudonyms independently from third party organizations. By carrying out related tasks directly on the devices the goal lay in minimization of potential user data concatenation.

Ten years later the first implementation recommendations for an IDMS followed [28]. The so-called *Identity Protector* was designed to protect privacy-sensitive data by using pseudo identities. The general idea behind those identities was comparable to today's digital identities (see Subsection 2.1.1). They were defined to hide users behind aliases. Rossum et al. also suggested to design IDMS solutions in a user-centered way, in order to be transparent and controllable. Users should have been allowed to decide whether to generate new pseudo identities for every transaction or communication partner or to reuse existing ones. The aim of this was to provide functionalities that reduce the amount of privacy-sensitive information sent to SPs.

These two approaches show that already a couple of years ago IM was an important aspect to deal with, even though internet and services at this time were not used as frequently as they are today. Without mentioning today's common terms researchers back then dealt with issues related to privacy disclosure. However, both addressed works present recommendations that are still partly applicable but have never been technically implemented. The steadily growing complexity of the internet and increasing numbers of services led to extensive and still ongoing research. Currently, different IDMS architectures are available. They all are distinguished by their implementation architecture design. One distinction that is most relevant for this project is presented next.

#### Centralized and Decentralized Architectures

IDMS designs in particular differ in their approaches to store identity and thus privacy-sensitive data. While some solutions follow *centralized* orientations, other systems prefer *decentralized* ones. Thereby, listed first architectures provide central storages that are used by large amounts of users together. Decentralized architectures on the contrary outsource identity data to user devices. Even though storing aspects are not the only distinguishing feature, others lie out of importance for this project.

Of course, both orientations have advantages and disadvantages. The most important ones for this project are listed in Table 2.1. They indeed show some key advantages in *centralized* over decentralized solutions. However, the project goal is the development of an IM and PM framework for *roaming users*. Thus, especially two of the presented advantages of *decentralized* solutions are the reason why the proposed framework is designed in this way. This is first, *high user reputation and trust* and second, *independence from central infrastructures*.

### 2.1.4 General IDMS Advantages, Disadvantages and Challenges

Subsections 2.1.4 and 2.1.5 are concerned with advantages, disadvantages and challenges of IM and IDMS solutions. The shown arguments are derived from extensive fundamental research and should be seen as extensions to the previous Table 2.1. It is also important that the Four Basic Framework Pillars within IM (see Subsection 2.1.2 and Figure 2.2) stay in tight relation to them.

The discussion is divided into two areas, according to the development focus on roaming users. The first one (Subsection 2.1.4) analyzes general aspects that are most relevant for traditional and

Table 2.1: Advantages and disadvantages of centralized and decentralized IDMS architectures

| | **Centralized** | **Decentralized** |
|---|---|---|
| **Advantages** | - potentials for high processing capabilities<br>- high physical security<br>- low risks of device lost or theft<br>- system maintenance by experts<br>- simple and well proven backup possibilities | - no need for connections to centralized infrastructure<br>- high user reputation and trust (personal device presents own area of responsibility)<br>- no need for user trust in central systems<br>- no need for (user trust in) central responsibilities<br>- probably cheap (minimal acquisition, no extensive protection of infrastructure) |
| **Disadvantages** | - requirement for connections to centralized infrastructure<br>- probably low user trust in systems and administrator<br>- need for expert knowledge to setup, run and maintain systems and infrastructure<br>- high processing capabilities come along with high costs (acquisition, integration and maintenance)<br>- requirement for highly protected infrastructure<br>- interesting for identity theft (large amounts of centralized data) | - limited processing capabilities on devices<br>- low baseline security (e.g. no company fences and walls)<br>- high risks of device lost or theft<br>- system maintenance probably by amateurs<br>- more difficult backup possibilities, if any |

currently used IDMSs (see Chapter 1). The second discussion (Subsection 2.1.5) then emphasizes on facts that are especially important when using mobile IM and IDMS solutions. With reference to Subsection 2.1.3 this separation also reflects the distinction between centralized (*general*) and decentralized (*mobile*) architectures. All listed advantages are comparable to potential user benefits of the project solution; they show added values of framework application into online identifications.

#### Advantages

The following list shows general advantages of IDMS solutions.

**Ease of Use and Efficiency** - Provide simplified ways to apply and effective and efficient ways for management and maintenance of identity data.

**Role and Context Dependency** - Role based orientation enables to choose best suitable identities within a particular context.

**User Release** - Support in remembering and submitting large amounts of identity data to avoid informational overload on user side.

**Transparency** - Process unification and standardization enables system and task transparency.

**User Responsibility** - User-centered system enable users with responsibilities for identity related tasks.

**Tracing** - Transaction logs, auditing and reporting functionalities allow to trace the life cycle of privacy-sensitive identity data.

**Detection of Security Breaches** - Provide mechanisms to detect system and security breaches and identity attacks (availability is solution dependent).

**Reaction to Security Breaches** - Provides mechanisms to automatically react to system and security breaches and to identity attacks (availability is solution dependent).

### Disadvantages

This list discusses disadvantages that (can) arise when working with IDMS.

**Personnel Requisition** - Need for highly skilled and trustable experts that are responsible to administer overall infrastructures and confidential and privacy-sensitive data.

**Administration Workload** - High workload and expenses for management and administration of systems and identity data.

**Costs** - Deployment, maintenance and training costs on a regularly basis.

**User Trust** - Centralized storages require high user trust[4].

**Third Party Dependency** - Potential infrastructure dependencies on and trust requirements to third parties for particular tasks (e.g. identity creation and validity checks).

**Attack Target** - Central data collections are particularly exposed to different kind of attacks.

### Challenges

Following, the most important challenges are shown that need to be considered when working with centralized IDMS systems.

**Security** - Need for secure data transmissions and storages (probably also back up).

**Availability** - Need for highly available systems and infrastructures.

**Conviction and Sensitization of Users** - Need to convince and sensitize users of advantages for taking part - IDMSs are only useful when well informed users entirely collaborate.

**Conviction of Third Parties** - Potential need to convince SPs and related third parties to participate and potentially adjust their services (solution dependent).

**User Interface Design** - Need for user-friendly and comprehensible interface designs in order to achieve user trust and high system and process transparency.

**Responsibility** - Need to find the balance between user responsibility and system automation.

**Deployment Strategy** - Need to choose the *right* solution. There are various systems and infrastructures available that are required to be taken into account. “Is the system just to solve a temporary problem or will it be used over a long time period?” [29]

[4]It is generally agreed upon that users mistrust central systems and prefer self-serviced solutions that lie in their personal area of responsibility.

### 2.1.5 Roaming User IDMS Advantages, Disadvantages and Challenges

As address previously, this subsection is based on the general advantages, disadvantages and challenges of Subsection 2.1.4 and adds information that is relevant for mobile IDMS solutions. However, many of the previous items are also applicable here. Suitable to this, Roussos and Patel analyzed project related issues that could occur when traditional IDMS approaches are applied into the mobility sector [30]. They came to the conclusion that this especially requires improvements within interoperability, privacy protection, security and self-configuration.

The most important challenges of mobile IDMSs are by far related to privacy issues. In some points privacy aspects can even be seen as their greatest disadvantages. That significance is thus the reason why related discussion is swapped into the separate Section 2.2; privacy does not get much attention in the following paragraphs. Apart from that, all shown aspects are essential for the upcoming development in order to achieve an efficient framework solution.

#### Advantages

The listed key advantages for IDMS solutions in mobile environments can be identified as potential user benefits of the proposed framework solution.

**Mobility** - Independence from central systems and infrastructures results in high flexibility.

**User Trust** - High user trust because no central administrator is required. Moreover mobile devices are mostly treated as personal belongings[5].

**Secure Storages** - Secure data storages can directly be integrated into the mobile devices.

**Standardized Communication Networks** - High availability of various standardized communication networks (e.g. Wireless Local Area Network (WLAN) and Universal Mobile Telecommunications System (UMTS)).

#### Disadvantages

When using IDMSs while roaming there are some disadvantages to deal with. Together with the following challenges they provide useful input for the upcoming design process because they show possible framework limits. Note, that privacy related disadvantages and challenges are very important but separately discussed in Section 2.2.

**Privacy Disclosure** - High risk to unnoticedly disclose privacy-sensitive information and especially locational based data[6].

**Device Theft** - Increased risk for device theft and loss requires baseline security on high levels and well structured emergency plans.

**Processing Capabilities** - Limited processing capabilities on mobile devices require appropriate mechanisms that still run smoothly (in particular relevant regarding cryptography).

**Application Installation** - Requirement to install appropriate user applications on the mobile devices.

[5] It is common that people trust their own technology more than *invisible* systems.

[6] More information regarding privacy disclosures and location based data is given in the addressed Subsection 2.2.

#### Challenges

The last discussion presents challenges of mobile IDMS systems.

**Overall Security** - Security aspects to deal with are more difficult than in well known, proven and widely accepted technologies as used in wired infrastructures and fixed environments.

**Baseline Security** - Application of the proposed framework is worthless if mobile devices are only secured on low levels or even not at all. Security for mobile IDMS infrastructures already starts with careful application of Personal Identification Number (PIN)s.

**Network Security** - The available communication networks are depending on the application partly not as secure as their wired counterparts (so far; esp. WLAN).

**Usability** - Usability aspects for interfaces are very important and difficult to deal with. It is necessary to reduce applications to small screens while at the same time presenting as much information as necessary in order to make processes and systems transparent.

**Supervision** - No central administrator is available to supervise correct handling. If users act wrong once, nobody notifies them and they most likely will keep on working in wrong ways.

**Conviction** - Convincing users and SPs to adopt not broadly used and unknown solutions can be extremely difficult; most parties are only aware of traditional IDMSs, if at all.

**Comprehensibility** - Together with usability aspects, all interfaces and processes require to be easy to understand, natural to use and comprehensible. An integration of work flows that users are already familiarized with, is highly supportive.

### 2.1.6 Conclusion

This section presented the theoretical background within IM. The provided information builds up the basis for the achievement of identity related objectives as listed in Subsection 1.2.2. It also contributes to answer research question one. The importance of the overall subject was substantiated by the description of two of the oldest system approaches and the growing demand for (mobile) IDMS solutions. This also visualized the project context once again. The discussed disadvantages and challenges point out potential framework limitations. It is thus important to keep them in mind during design and development.

The following list concludes this section. It is based on [29] and presents seven potential IDMS categories. The differentiation was defined in the end of 2003 by Chris Pick who is the vice president of market strategy for NetIQ Corp.[7]. It helps to get an insight into the large field within IM and IDMS and to visibly place the contribution of this work.

1. **Authentication** - Proving who you are and what you have access to.
2. **Directory Administration** - Providing secure administration of directory enabled products within the enterprise.
3. **Single Sign On** - Accessing multiple applications through one authentication agent.
4. **User Provisioning** - Enabling management of users from day one to the day they part from the organization.

[7] NetIQ is a global leader in systems and security management. See `http://www.netiq.com/`.

5. **Password Management** - Providing abilities to self service password resets.
6. **Extranet Access Management or Web Single Sign On** - Supporting authentication through a web single sign on portal that provisions entitlements to other applications users have access to.
7. **Delegated User Administration** - Consolidating redundant manual processes for direct and centralized management.

According to these categories, the overall project subject can be seen as a composition of system parts belonging to the *authentication*, *single sign on*, *user provisioning* and *delegated user administration* IDMS technologies.

## 2.2 Privacy Management

This section adds project fundamentals by discussing the working area within privacy. Like the previous section this one also starts with identifying the most relevant terms and definitions. This is followed by explanations of important privacy issues that need to be faced during framework development. Next, technology design fundamentals and guidelines are presented to tighten the direction of the upcoming design process. This information also helps to visualize and clarify the overall solution approach. Five essential PM design pitfalls round up the first part of this section.

The second part begins with a presentation of technology design fundamentals for the development of PM systems. Together with particular design guidelines they are used to specify important PM Framework Design Requirements for the upcoming development. A description of a sample service request with embedded PM functionality concludes Section 2.2.

### 2.2.1 Terms and Definitions

This subsection briefly shows the terminology within PM that is important for the carried out project. It discusses definitions of the terms "Privacy", "Privacy Management" and "Pseudonymity and Anonymity". For reasons of clarity the structures of Subsections 2.1.1 and 2.2.1 are the same.

#### Privacy

| **Privacy** *The right to be let alone and the claim of individuals to determine for themselves when, how, and to what extent information about them is communicated to others.* |
|---|

So far various definitions regarding the privacy concept have been published. Probably one of the most common was announced by Warren and Brandeis in 1890. In response to the rising development in printing technologies and the increasing amount of published newspapers and photographs[8] they described privacy as "the right to be let alone" [31]. Warren and Brandeis considered that in those times something was necessary to protect privacy-sensitive information from being "shouted from the rooftops". A further familiar definition was published by Westin in 1967. Back then he

[8]For more information it is referred to "Privacy Law in the USA" at `http://www.rbs2.com/privacy.htm`.

characterized privacy as "the claim of individuals, groups or institutions to determine for themselves when, how, and to what extent information about them is communicated to others" [32]. Privacy in the project context particularly aims to manage and restrict access to identity data.

#### Privacy Management

**Privacy Management** *Operational and technical functionalities and tools that support users to meet privacy requirements when they deal with privacy-sensitive data.*

"PM involves the strategies and safeguards used to protect the privacy [...]. Safeguards are enforced so that information cannot be released to or accessed by unauthorized subjects." [33] This definition of the general idea behind PM was published by the American Institute of Certified Public Accountants (AICPA)[9]. PM specifies ways to consciously handle privacy-sensitive data. It provides operational and technical functionalities and tools that help users to fulfill privacy requirements. In collaboration with IDMSs PM supports users during the entire identity life cycle (creation, maintenance, termination). It enables ways to ensure the right that information stored in IDMSs and on SP sides is valid and up to date.

#### Pseudonymity and Anonymity

**Pseudonymity and Anonymity** *Pseudonymity is used to obscure identities behind aliases. The better connections between aliases and identities are hidden the higher is the degree of achieved anonymity.*

Pseudonymity and anonymity are technical ways to support the protection of privacy-sensitive data. By obscuring real identities behind aliases it is possible to conduct transactions without revealing unnecessary personal and privacy related information. Like in real life the (amount of) published facts about oneself depends on the situation. Using context dependent pseudonyms that combine different identity attributes n online identifications thus allows to work private. In addition, it is common to use the term *anonymity* to classify a pseudonym's quality or strength. "Anonymity states that an individual should not be identifiable within an anonymity set, that is, a set of users." [34] That means, that pseudonyms which highly prevent identification of individuals result in high anonymities[10].

### 2.2.2 Privacy Issues

Chapter 1 mentioned that this project treats PM more as system requirement rather than handling it as an independent subsystem. Therefore, general and roaming user advantages, disadvantages and challenges (as presented within the IM section) are intentionally left out. Instead, it is much more important to show major privacy issues to face during development. This establishes first impressions of PM as part of the framework and roughly visualizes system requirements.

[9] AICPA is the national, professional association of Certified Public Accountants. It's main goal is to define rules and to set standards for the information technology. For information see `http://infotech.aicpa.org/`.

[10] In high level protections identification is only possible with disproportional expenses of time, money and work.

**User Unawareness**
Searching for the most obvious privacy issue by far automatically results in blaming users. The majority of them is not aware of how much personal information they unknowingly share. They mostly do not even recognize consequences of privacy disclosure. To work against this essential fundamental issue training on regularly basis is required. Users have to be sensitized to the importance of informational privacy.

**Identity Theft**
According to [35] the frauds of identity theft and misuse are increasing. Identity thieves try to get access to resources on the basis of stolen identities. Two important ways to keep this risk low are to minimize disclosures of personal information and to protect data storages and transmissions.

**Profiling and Concatenation**
SPs often create user profiles. If users reveal more information than necessary they strongly support this process. But even if they disclose only minimum sets of identity attributes, SPs are still able to concatenate new and old information. In this way SPs are able to gather important facts that users probably think they have never disclosed. Using pseudonyms whenever possible is the first step to reduce dangers of this privacy issue. However, users also need support from the IDMS to identify profiling and concatenation risks.

**Locational Disclosure**
More and more SPs publish services that are based on locations of requesting users. While those services may look useful in first place they come with a large privacy issue. Dependent on the refresh period, users are required to disclose current positions once or even permanently over longer times. Especially the fact of location tracking harms privacy to large extents. Thus, the same as for the previous issue, it is important to minimize disclosures of locational data whenever possible.

**Data Misuse and Access**
One essential issue related to collected data is that users mostly do not have any possibility to ensure that SPs handle appropriately. Even if particular specifications were published, there is no way for user verification. Besides this, users are mostly prohibited from accessing collected data, so that they are not able to validate, modify or delete disclosed attributes. Here, IDMSs are required to support users best possible. They have to allow mechanisms to ensure correct handling and to enable simple ways to access collected data.

Concluding the overview of the most important privacy issues, two quotations are presented. They emphasize on the difficulties when facing and resolving privacy issues. The company Grid-Tools thus stated that "the challenge in data privacy is to share data while protecting personally identifiable information." [36] In addition, Goerlach, Heinemann, and Terpstra pointed out that "perfect privacy is clearly impossible as long as communication takes place" [37].

### 2.2.3 Technology Design Fundamentals

In combination with the next subsection this one discusses the technology designs of PM solutions. Subsection 2.2.3 starts with describing significant design fundamentals. It presents five pitfalls that are important to avoid when developing privacy functionalities. The subsection then identifies general accepted design goals for integrating those functionalities into other systems. Taking both presentations as references help to develop a well designed, structured and partly standardized PM framework component.

### Five Design Pitfalls

In their paper [38] Lederer et al. analyzed various system infrastructures that are concerned with processing privacy-sensitive data. The study aimed to identify different approaches to integrate PM functionalities. Based on their results the authors analyzed five design pitfalls that can be seen as "guidelines for designers of privacy affecting interactive systems". As a reason that the user tool as part of the framework solution matches this statement the pitfalls have a high impact on this project. The following list quotes the original terms and presents short descriptions and hints on how to avoid each pitfall.

**1. Pitfall: Obscuring Potential Information Flow**
An effective, efficient and transparent use of privacy related systems can only be achieved with well informed users. They need to be aware of *potential* privacy disclosures through used systems. This helps to identify "social consequences of its use". Users furthermore are required to know the systems' capabilities and limits within privacy. Therefore, systems need to provide information about every process unit and each working step. It is necessary to clarify the kind of collected data, possible parties that are allowed to process it, the way (the media through which) it is collected, the duration of storage and the potentials for unintentional disclosures.

**2. Pitfall: Obscuring Actual Information Flow**
While the first pitfall emphasizes on obscuring *potential* disclosures, this aims to reveal *actual* ones. Users need to be permanently informed about privacy disclosure during runtime. No critical action is allowed in a hidden way. Thus, users are provided with a more transparent view over the entire system and carried out processes. Whenever disclosures occur users need to be informed in an *obvious* way. "If this is impractical notice should be provided within a reasonable delay." Feedback also helps to avoid this pitfall; continuously notifying of current processing stages enables high system transparency.

**3. Pitfall: Emphasizing Configuration Over Action**
According to Palen and Dourish "setting explicit parameters and then requiring people to live by them simply does not work" [39]. Systems rather need to be designed in such a way that privacy regulations are "embedded components of the activity". That means, excessive steps to create and maintain privacy are to be avoided without direct relation to currently carried out activities. This is important, because it has been broadly analyzed that those configuration steps will not achieve desired behaviors. Privacy related specifications need to be a "natural consequence of ordinary use of the system". Generally, whenever possible, privacy configurations need to be embedded into activities and prevented from being extracted into separate mechanisms.

**4. Pitfall: Lacking Coarse Grained Control**
PM designs need to provide "obvious, top-level mechanisms for halting and resuming disclosure". These mechanisms to enable process cancellations can for example be realized by self descriptive buttons or switches. The goal is to allow users to abort situations in that the fine grained preferences of pitfall three are not set properly; so users always have a way out.

**5. Pitfall: Inhibiting Existing Practice**
The fifth and last pitfall is not as important for the framework development as the others. It defines that systems should not prohibit users from applying existing, established and well known practices. This is the fact, because it is generally much easier to understand requirements if it is possible to integrate processes and tasks that oneself is already familiar with.

These pitfalls, and especially the first four, show the importance for handling PM developments carefully. Keeping them in mind during framework design and development helps to avoid stepping into already researched issues. Moreover, effectively bypassing them enables achievement of a partly standardized solution.

#### Design Goals

Related to the pitfalls, four commonly agreed goals exist for systems designs that enable privacy functionalities. As natural, there is some overlap.

**User Driven Control**[11] - Users need to be enabled with full control over their personal data. This means, that they have to be allowed to decide critical process steps. Furthermore, they have to be provided with ways to abort processes or entire transactions.

**Anonymity** - Systems are required to support users in applying best suitable pseudonyms. Furthermore, users also need to be allowed with ways to choose anonymity levels.

**Informed Consent** - Systems are required to work with informed user consents. That means, processes are to be transparent and users need to actively agree on data disclosures.

**Logging and Tracking** - As already mentioned as IM goal (Subsection 2.1.2), logging and tracking are essential to provide high degrees of transparency and to enable reviews.

### 2.2.4 Technology Design Guidelines

The previous subsection showed PM technology design fundamentals. Based on that information Subsection 2.2.4 presents two important PM design guidelines. Those are the *Informational Self-Determination* and the *Fair Information Practice Principles (FIPPs)*. They both are widely known and accepted as references for PM infrastructure designs. Concluding Subsections 2.2.3 and 2.2.4 a summary points out consolidated PM design requirements.

#### Informational Self-Determination

As briefly addressed previously, *Informational Self-Determination* is a term published by the German Federal Constitutional Court during the census in 1983[12]. It specifies recommendations that are concerned with automated collections of personal information, initiated by digital systems. Those recommendations emphasize on the importance of user-centered responsibilities when working with personal data. This helps to avoid unintentional data disclosures and processings. The objectives of the underlying approach are best described by quoting the Constitutional Court itself [41][13].

[11] This particular design goal is also described as *self-determination*.

[12] For more information about the 1983 census in Germany see for example the paper *"Data protection in Germany I: The population census decision and the right to informational self-determination"* published in [40].

[13] The original version of the census was written in German language. However, the quotes are commonly accepted English translations.

> The individual [...] has the right to know and to decide on the information being processed about him. [...] In the context of modern data processing, the protection of the individual against unlimited collection, storage, use and disclosure of his/her personal data is encompassed by the general personal rights of the [German Constitution]. This basic right warrants in this respect the capacity of the individual to determine in principle the disclosure and use of his/her personal data. Limitations to this Informational Self-Determination are allowed only in case of overriding public interest.

This information briefly introduces that particular design guideline. Information Self-Determination can thus be seen as the main goal of IDMS solutions as soon as privacy functionalities are integrated. In those cases we also speak of Privacy Enhancing Technologies (PET). However, as a reason that PET is separately discussed in the next subsection, those few paragraphs here already close the related discussion. Further information is then rather given in Subsection 2.2.5.

#### Fair Information Practice Principles

A second very important reference within privacy system design is provided by *FIPPs* [42]. FIPPs was communicated by the United States Federal Trade Commission[14] in 1998 in response to the increasing amount of automated data systems dealing with privacy-sensitive data. Today, FIPPs is used as the "basis of many modern international privacy agreements and national laws" [43]. Even though the definitions are especially relevant for organizations in the United States, they also supplement the framework development process in a wide extent.

The entire FIPPs' concept is based on five core principles. They describe how to appropriately handle privacy-sensitive data in IDMS solutions. Their goal is to ensure fairness, privacy and security. FIPPs is widely accepted in electronic transmissions that are concerned with processing such kind of data. It is also already integrated into a few IDMS solutions, like for example [44]. The following description about the five core principles is based on [42].

**1. Principle: Notice and Awareness**
Users need to be informed about SPs' data management practices before any information is collected. This notification has to be presented easy to read and understand. Important facts are the collecting entity, uses to which collected data will be put, potential recipients of collected data, whether provision of requested data is voluntary or required (including consequences of declines) and steps taken to ensure data confidentiality, integrity and quality.

**2. Principle: Choice and Consent**
Users need to be provided with different possibilities to choose on how personal data may be used after collection (internally or externally). This can be achieved by allowing opt-in or opt-out selections[15]. To improve accuracy for accepting or declining processings, consents can be tailored to more detailed decision areas.

**3. Principle: Access and Participation**
Users have to be given ways to access collected data. This enables reviews, provides possibilities to

[14] Federal Trade Commission is an independent agency of the United States government. Their mission is *consumer protection*. For more information visit `http://www.ftc.gov`.

[15] *Opt-in*: Users need to explicitly provide consent. If no action is made, SPs assume that no consent is given. *Opt-out*: Consents are given by default. Active decline is required (e.g. removing checked box).

verify accuracy and completeness and supports information of SPs of any changes to stored data. It is very important that access is timely and inexpensive.

**4. Principle: Integrity and Security**
SPs need to assure that collected data is accurate and secure. Data integrity can be supported by cross referencing against multiple reputable sources. And again, users need ways to access and verify data. Untimely information is required to be deleted or converted to anonymous data on a regularly basis. For security reasons SPs are responsible to protect collected data against loss, unauthorized access and use, destruction and disclosure. This also includes encryption of data storages and transmissions.

**5. Principle: Enforcement and Redress**
SPs need to ensure enforcement of their published privacy protection principles. Three common enforcement types are *Self Regulation* (comprised of collectors themselves or appointed regulatory bodies), *Private Remedies* (responsible for individual civil causes of action after data misuse) or *Government Enforcement* (responsible for civil and criminal governmental penalties).

While all five principles are consistent in themselves, there is still one disadvantages of the entire FIPPs concept - the principles are only recommendations; they do not ensure enforcement like for example by laws. The principles are rather primary based on self regulation. However, they are still commonly applied and adapted as baselines for several other privacy frameworks. This leads to the conclusion, that FIPPs and the Informational Self-Determination provide valuable, standardized design guidelines for the upcoming framework development within privacy.

### Consolidated Privacy Management Framework Design Requirements

The following list sums up the most important aspects belonging to the Technology Design Fundamentals of Subsection 2.2.3 and the Technology Design Guidelines of Subsection 2.2.4. It also adds valuable information analyzed by Hyppönen in his work [45]. As briefly introduced in the subsection's ingress, this list aims to specify privacy related framework design requirements. For referencing reasons they are abbreviated as *PDR*.

- **PDR 1. Data Minimization** - Reduce disclosed privacy-sensitive information to a minimum.
- **PDR 2. Ease of Access and Revocation** - Provide simple and timely ways for users to access, verify, modify and revoke collected data.
- **PDR 3. Security** - Ensure secure storages and transmissions of privacy-sensitive data. Avoid unauthorized access and changes to this data, to ensure confidentiality and integrity.
- **PDR 4. Logging** - Integrate mechanisms to log transactions and processes. Ensure that logs are easy to locate and access. Allow both, users and system to review logs.
- **PDR 5. Pseudonymity and Anonymity** - Provide possibilities to apply pseudonyms in different levels of anonymity, wherever possible. Support users in choosing the best suitable level and allow them to manually adjust system choices.
- **PDR 6. Unlinkability** - Try to ensure that multiple service requests and transactions can not be linked to each other. Avoid ways of user profiling on SP sides or at least warn users accordingly.

**PDR 7. Untraceability** - Try to ensure that locational information can not be used for position tracking or at least warn users accordingly.

**PDR 8. User Centering and Transparency** - Carry out processes and transactions as transparent and user-centered as possible. Provide continuous feedback on the current system status. Allow users easy ways to abort tasks, transmissions and entire transactions.

**PDR 9. User Consent** - Ensure that exchanges of privacy-sensitive data are based on well informed users. Avoid sending data without user consents.

### 2.2.5 Technology Architecture Design Approaches

Similar to Subsection 2.1.3 of the IM discussion this subsection points out two fundamental PM related technology architecture design approaches; PET and P3P. The PET term is already mentioned a couple of times previously. This subsection now clarifies the idea behind it. Related design principles then emphasize on the connection to the Information Self-Determination and the framework development. Describing P3P as the second technology approach further illustrates the development direction. Finally, a typical service request with embedded P3P functionality is described. This typical process sequence has great impact on the project because the design of almost all privacy related framework mechanisms originates from it.

#### Privacy Enhancing Technology

PET is the most often used term when embedding PM functionalities into IDMS infrastructures. According to Borking and Raab [46] PET can be described as follows.

> [...] a coherent system of Information Communication Technology measures that protects privacy by eliminating or reducing personal data or by preventing unnecessary and/or undesired processing of personal data; all without losing the functionality of the data system.

There are two primary goals when integrating PET into IDMS infrastructures. Both emphasize on a tight connection to the PM Framework Design Requirements of Subsection 2.2.4. The first is again the support to manage and maintain entire identity life cycles in user-centered ways. The second is to precisely notify users about currently requested identity attributes in an easy and comprehensible format. This helps to disclose only minimum sets of identity attributes and to recognize potential privacy risks in advance.

That information so far does not provide any essential new knowledge for this project. However, there are five important aspects that are nowadays commonly considered to be the *Principles of PET*. Those were initially collected by Hansen in [47] and later on summarized by Hansen et al. in [48]. They prescribe *Data Minimization*, *Transparency*, *System Integration*, *User Empowering* and *Multilateral Security*. While the first two principles align with the aforementioned privacy requirements, the other three provide additional specifications. Therefore, those are added as PDR 10, 11 and 12 to the PM Framework Design Requirements of Subsection 2.2.4.

**PDR 10. System Integration** - Integrate privacy protection functionalities directly into the systems and underlying infrastructures. Avoid additional implementations or processes.

**PDR 11. User Empowering** - Provide system driven, automated selections of best suitable privacy levels and allow users to adjust those choices.

**PDR 12. Multilateral Security** - Work with a requirement of minimal trust relations in other parties.

### Platform for Privacy Preferences

The previously described PET approach can be seen as the goal when integrating privacy into IDMS infrastructures. However, privacy protection already starts in a much earlier stage. Thus, before any transaction is initiated, users and SPs need a way to agree on applied data management practices. On this background, the *P3P technology* was created. It was initially published by W3C in the mid 1990s and from there on continuously enhanced and improved. Even though further development was paused in the end of 2006, the functionality can still be applied and independently extended in it's current and final state. Moreover, the developers promised to resume research as soon as there is demand for improvements.

The P3P vision is to "increase user trust and confidence in the web through technical empowerment" [49]. It is said, that "P3P enabled web communication can bring ease, transparency and consistency to web users wishing to decide whether and under what circumstances to disclose personal information". P3P is comparable to human readable privacy policies. However, the problem of this *traditional* approach is, that those written guidelines are often not easy to locate, complicated to read and even harder to entirely understand. To overcome these issues, P3P automates related processes. It helps to offload users in making privacy decisions by letting the systems automatically work for them, based on personal demands. Therefore, SPs publish P3P policies that define applied data management practices. Those specifications are made with predefined P3P vocabulary, syntax and formats. In the default P3P configuration SPs are able to provide answers to the following project relevant questions (based on [49]).

- Who is collecting data?
- What data is collected?
- For which purposes is the data collected?
- Who are the recipients of the collected data?
- Which collected data can be accessed by users?
- What is the data retention policy?
- How will disputes about the policy be resolved?
- Where is the human readable privacy policy located?

While in the first impression the general idea behind P3P reflects privacy related framework goals, there is still a challenge to face: P3P is primary designed for fixed environment. Particularities of the mobile area, for example locational based identity attributes, thus make it necessary to extend the default vocabulary. For exactly this, the developers integrated an <EXTENSION> element. This enables to modify the aforementioned questions, add new ones or even adjust P3P vocabulary and syntax. However, technical specifications for that particular task are not part of the current discussion and rather treated in Chapter 3.

To provide a better understanding of the P3P technology, a typical process sequence for requesting a P3P enabled service is described and then visualized in Figure 2.3. It is based on the communication partner *User* who requests a service and *SP* who provides it.

1. *User* requests the P3P policy from *SP*.
2. *SP* responses with requested policy.
3. On *User* side: An application automatically interprets the P3P policy. It verifies whether *SP* data management practices comply with *User* requirements[16] or not. If they contradict, the application notifies *User*, so that he can act accordingly. Otherwise step 4 follows.
4. The application on *User* side initiates the service request.
5. *SP* replies with the response and provides *User* with the requested service. Now, the entire transaction is based on data management practices that are agreed upon on both sides.

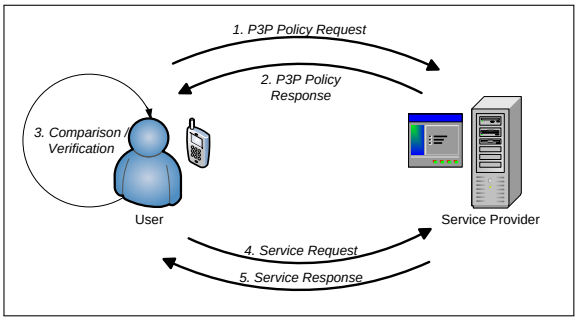



Figure 2.3: Service request with embedded P3P functionality (based on [50])

As a conclusion, P3P comes with two limitations. First, it is not able to enforce compliance with published data management policies. It only tries to assure agreements between SPs and users regarding applied data management practices. Second, P3P is not responsible to protect data storages and transmissions. However, a few PET enabled IDMS infrastructures already demonstrated, that P3P provides great benefits (e.g. [6], [12] and [16]). Moreover, the *Independent Center for Privacy Protection Schleswig-Holstein, Germany*[17] stated that they plan to establish P3P into a broader context. For this, they are working on enforcement methods and default P3P policies that comply with the European data protection regulations.

### 2.2.6 Conclusion

In the same manner like Section 2.1, this one presented the theoretical background of this project. By analyzing the research area within privacy, it rounds up the project foundations. The key fact been worked out, is to design the framework user-centered. Only this way enables users with full control over their privacy-sensitive data. However, it is important that this *full control* is not limited to data storages and applications. Users rather need ways to access collected data on SP sides. This helps to verify, modify and delete identity attributes.

[16] This verification is achieved by comparing the SP policy with other policies that represents the User's privacy specifications. These particular policies define data management practices that User allows to be applied by SP.

[17] The Independent Center for Privacy Protection in Germany is responsible to control data protection. Fore more information it is referred to `https://www.datenschutzzentrum.de/p3p/` (in German language only).

The discussed privacy issues and the technology design fundamentals and guidelines are valuable inputs for the upcoming framework development. Especially the first four pitfalls help to overcome well researched design issues within privacy. Furthermore, the PM Framework Design Requirements consolidate all privacy requirements that were mentioned throughout the entire Section 2.2. If the framework fulfills them, high privacy can be achieved. Finally to say, Køien and Oleshchuk accurately remarked the following, regarding the development of privacy concerned systems [51].

> There clearly is a trade off between safety/security and privacy in the sense that we may have to give up some personal privacy in exchange for better safety/security. [*It is important to find*] a balance between surrendering our privacy versus benefiting in terms of improved user convenience.

## 2.3 Background Summary

In this chapter the project foundations within the two treated working areas of identity and privacy were established. Important terminology and design fundamentals, guidelines and goals were presented. Moreover, interesting system architecture approaches for both fields were stated.

The first section in particular emphasized on general accepted design goals for systems that deal with privacy-sensitive identity data. Those goals were used to derive Four Basic Framework Pillars within IM. Keeping their balance is essential for a standardized and stable overall solution. A presentation of two former architecture approaches then showed that research in the identity area already started a couple of years ago. It also depicted that there is a steadily growing demand for IM functionalities. Next, it was identified why the intended framework solution is based on a decentralized rather than a centralized architecture approach. Section 2.1 also contributed to answer the first research question. Thus it discussed advantages, disadvantages and challenges of general and mobile IDMS solutions. This showed the direction of the upcoming framework development and aimed to make user benefits and added values of the solution clear.

In the second section especially the description of privacy issues has an important impact on the following requirements specification. The presentation of technology design fundamentals then visualized and clarified the intended overall solution approach. This section also worked out two well known design guidelines for privacy related system developments. Furthermore, one of the most essential discussions of Section 2.2 established twelve PM Framework Design Requirements. Those are great guides for privacy developments. The section then was concluded by a presentation of a sample service request with embedded P3P functionality. As it is seen shortly, this particular process sequence is the origination of all framework aspects within privacy.

With reference to the development strategy of Figure 1.1, the following can finally be stated. Now, the foundations within the research areas of IM and PM have been established. The background research for this project is thus completed. This means, that the next process steps are the definition of requirements and specifications for the framework solution and the design and development according to them. These process steps are part of the following Chapter 3. That one is also introduced by a further summarization of the most important points of this chapter.

# CHAPTER 3

# Framework Development

Chapter 3 presents the solution. According to Figure 1.1, it covers requirements and specifications, design and development steps. Even though transitions between the framework parts - *procedural method*, *privacy and security protocols* and *user tool* - are fluid, it is attempted to classify related tasks and mechanisms in the best possible way. While short conclusions in the end of each section review the most important points, further interim conclusions are added whenever necessary. This chapter shows figures, tables, UML diagrams, process sequences and system architectural designs. This allows a direct handover of the presented design to programmers, in order to build first prototypes.

Section 3.1 starts with a brief review of the background research to establish basic development foundations. Typical user scenarios then present problems, challenges and goals to be covered in the project solution. The scenarios are also useful for focusing on user benefits.

For ease of reading, the following two sections are structured in the same way. Section 3.2 gives a detailed requirements analysis to specify all three framework parts in a clear and detailed manner. Moreover, first draft requirements for underlying infrastructure and applied technology are presented. Last but not least, six functional requirements for the overall solution are worked out.

Section 3.3 is based on those requirements. It presents the related framework design and development process that is carried out. Different implementation ways are shown and references to other works are established. This section also specifies additional system units that are required to place the framework solution in the intended context.

The last section introduces Chapter 4 and summarizes the entire design process and the framework solution with emphasis on the overall technical system architecture and reviews user scenarios and functional requirements. Moreover, it shows requirements for SPs and clarifies the specifications for applied infrastructure and technology that were drafted in Section 3.2.

## 3.1 Basic Foundations

Section 3.1 combines the background research of Chapter 2 with the upcoming design process. It briefly reviews the development fundamentals, design requirements and the overall development goals. This section also helps to once again point out the context in which the framework is placed.

Two sample user scenarios show typical identity and privacy related process sequences. To visualize the motivation for the chosen implementation design, they are taken into account in different process steps. Last but not least, a couple of framework related scenario questions point out user benefits of the proposed overall framework solution for IM and PM on mobile devices.

### 3.1.1 Review of Framework Specifications

With reference to Chapters 1 and 2, Table 3.1 (pages 37ff) reviews the most important specifications and requirements for the framework design. It also serves as a discussion base in Chapter 5. The table shows the non-functional requirements, the Four Basic Framework Pillars within IM, the PM Design Pitfalls and the Privacy Design Requirements. The last row presents the chosen PM Design Guidelines that are widely accepted references when developing privacy related systems. Because of limited space, abbreviations are partly applied, but the *Section* column indicates where the particular terms were introduced, so that more information can be retrieved.

Table 3.1: Review of framework specifications

| **Term** | **Content** | **Section** |
|---|---|---|
| Non-Functional Requirements | NFR 1. Executable on today's mobile phone platforms<br>NFR 2. Support of different privacy, anonymity, accountability and confidentiality degrees<br>NFR 3. Application of standardized protocols<br>NFR 4. Easy to understand and simple to use tool<br>NFR 5. Specifications for SPs | 1.2.3 |
| Four Basic Framework Pillars within IM | Security<br>User Trust<br>Cost Efficiency<br>Ease of Use | 2.1.2ff |
| PM Design Pitfalls | Obscuring Potential Information Flow<br>Obscuring Actual Information Flow<br>Emphasizing Configuration Over Action<br>Lacking Coarse Grained Control | 2.2.3ff |
| Privacy Design Requirements | PDR 1. Data Minimization<br>PDR 2. Ease of Access and Revocation<br>PDR 3. Security<br>PDR 4. Logging<br>PDR 5. Pseudonymity and Anonymity<br>PDR 6. Unlinkability<br>PDR 7. Untraceability<br>PDR 8. User Centering and Transparency<br>PDR 9. User Consent<br>PDR 10. System Integration<br>PDR 11. User Empowering<br>PDR 12. Multilateral Security | 2.2.4ff |
| PM Design Guidelines | Informational Self-Determination<br>Fair Information Practice Principles | 2.2.4ff |

### 3.1.2 Typical User Scenarios

In this subsection two sample user scenarios are introduced. They describe typical process sequences that can be supported by IM and PM systems. These scenarios are referenced throughout the entire chapter to highlight user benefits and to point out the motivation for chosen implementation directions. Moreover, they also visualize the solution context.

**Scenario 1: Registration and Participation in Online Bulletin Board**
*DressYourClothes*[1] discovered that many people are missing a way to discuss clothings on the internet. Therefore they developed an online bulletin board. To avoid misuse, it requires registration with username, password, email and postal address and telephone number. To cover running expenses, DressYourClothes integrated an opt out checkbox into the registration form. Selected by default, it allows to use provided data for marketing purposes.

**Scenario 1** *Carol wants to sign up for the bulleting board of DressYourClothes. During registration she is asked for username, password, email, postal address and telephone number. She wants to start a discussion and feels bothered by the large amount of requested data. Not concentrated, Carol quickly provides every information box. She never reads the General Terms and Conditions (GTC) and therefore automatically checks the box to accept them. Carol ends the registration by clicking the submit button. Unfortunately she missed the ticked marketing checkbox.*
*A couple of days later Carol wants to login again. After thinking for a while she remembers username and password. In the meantime, Carol received plenty of phone calls and lots of advertisements by email and mail. Strangely enough, she never gave away personal identity data to companies that are contacting her now. Even worst, some of the received brochures and fashion catalogs are actually personalized for Carol. She is very unsettled and does not know which mistake she did.*

**Scenario 2: Online Shopping**
The online store *YouBuy* distributes concert tickets and merchandise articles; user registrations are optional. To continuously improve service, YouBuy added a survey to their ordering processes. It voluntary asks customers for further personal details that are not necessary to complete shopping. However, to motivate customers a discount for next purchases is awarded to all participants.

**Scenario 2** *Leon browses to YouBuy's online shop and navigates through the different categories. He spots his desired event and places a discounted ticket for students into his shopping basket. After some time Leon decides that the shopping basket contains all desired items. He clicks on the checkout button to complete the order.*
*As a reason that Leon does not have a personalized user account, YouBuy asks for postal address and some further details that are necessary to receive the student discount. Leon fills in all necessary data and submits them. While printing the booking confirmation he is automatically confronted with a survey. Normally he never takes part in polls, but this time YouBuy lures him with a discount. Leon really wants to get this and therefore answers the questions. Without any hesitation he provides additional personal details.*

### 3.1.3 Conclusion

Concluding the review, seven framework and user scenario related questions are stated. They are typical when working with identity and privacy systems and can be used to show the direction of the framework solution. This is based on problems that need to be carefully considered during requirements and specifications. In order to directly point out appropriate issues, the most relevant Privacy Design Requirements (Table 3.1) are listed in parentheses.

1. How can Carol and Leon be supported to fill in identity attributes and to reduce the workload for applying the same data in subsequent transactions? (*PDR 1, 8, 11*)

[1] All company and user names are just fictitious to simplify the presented scenarios.

2. How can Carol and Leon be informed about the transaction's potential level of privacy, privacy risks and issues and third parties that are allowed to process collected data? (*PDR 6, 7, 8, 9, 12*)
3. How can Carol and Leon be enabled with simple ways to review and modify disclosed personal identity data? (*PDR 2*)
4. How can Carol and Leon be allowed to review past transactions? (*PDR 4*)
5. Which system infrastructure requires minimal trust relations in other parties? (*PDR 8, 12*)
6. How to securely store privacy-sensitive data on Carol's and Leon's mobile phones, and how to guarantee protected and anonymous transmissions? (*PDR 3, 10*)
7. How to support Carol and Leon to disclose only a minimum set of personal data, and to what extent can anonymity be guaranteed? (*PDR 1, 5, 11*)

## 3.2 Requirements and Specifications

This section carries out the framework requirements analysis and identifies the functional requirements. A brief preparation that defines underlying infrastructure and technology is necessary to achieve smooth integration of the solution into a larger context. It also provides a good overview of the upcoming analysis process. Considering the two typical user scenarios, the Framework Pillars within IM and the Privacy Design Requirements, the *procedural method*, *privacy and security protocols* and *user tool* are specified. While the goal of the current section is to define framework requirements, Section 3.3 presents their technical implementation. However, some intersections and overlappings are possible.

### 3.2.1 Requirements Preparation

Aiming at a broad acceptance and compatibility, the framework tries to restrict underlying infrastructure and applied technology as little as possible. However, entire independence is impossible. Thus, the next paragraphs anticipate the upcoming analysis and briefly describe key requirements for those working areas. The main goal is to provide a brief overview; all specifications are then clarified in the following discussions.

**Mobile Phone Platform**
Even though no prototype is developed, user tool interfaces and mechanisms are exclusively discussed and designed for the Android mobile platform. Compatibility proofs for other platforms are out of the project scope.

**Underlying Communication Infrastructure**
Analysis of the underlying communication infrastructure resulted in the conclusion that no particular specifications or limitations apply. The only requirement is that Transmission Control Protocol / Internet Protocol (TCP/IP) data streams can be interpreted and processed in order to receive and forward data packets in the proposed anonymous network. Thus, the two mostly used mobile communication networks UMTS and WLAN are entirely supported.

**Encryption Protocol**
The user tool integrates appropriate mechanisms to split, encrypt, decrypt and merge data packets.

This is crucial to successfully work in the designed anonymous network. But SPs are currently required to apply a particular protocol because so far no comparable standardized solution is available. Only with this protocol they are able to carry out framework tasks accordingly.

**Location Based Services**
The anonymous network hides sending stations from SPs. Therefore, locational based services are not usable in their default ways (see Subsection 2.2.2). However, in order to keep the support, SPs are required to include corresponding specifications into their data management policies. The largest advantage of this is that disclosure of positional data not longer takes place unknowingly. The addressed policies as far as their technical integration are discussed in Subsection 3.3.1.

### 3.2.2 Procedural Method

For the requirements analysis it is essential to define the overall procedural framework method that shows the proposed way for using mobile devices in privacy-friendly online identifications. The discussion starts with a presentation of the basic idea that is followed by more detailed considerations. A short review then concludes this section.

#### Basic Idea

The overall idea of the proposed framework for IM and PM on mobile devices is that *user tool* (or in short *tool*) comprises the main system component on mobile phones. It provides a browser to anonymously surf the web. The key feature is that it automatically recognizes identity attribute requests and allows users and SPs to agree on data management practices before any data is transmitted. To unburden users, it then automatically selects requested data from internal databases that are securely stored on the mobile phones rather than in central storages. The tool aggregates chosen attributes and in cases users agree, securely and anonymously transmits those pseudonyms. Finally, the user tool interprets SP response messages and displays results on the mobile phone screens.

Figure 3.1 illustrates this sequence in a simplified UML diagram. As a reason that the activity of surfing the web is unimportant for now, it only presents steps that are applied in cases identity data is requested. This figure emphasizes on the two user tool components *Identity Agent* and *Identity Proxy*[2] and separates the entire sequence into four process steps. This division is applied in order to simplify identification of required protocols and user tool functionalities. In the next paragraphs each process step is described in more detail.

#### Detailed Procedure

According to Figure 3.1 every transaction consists of four key steps. While *Process Step 0: Initial Configuration* is mostly carried out once, *Process Step 1: Data Management Practices Agreement and Privacy Identification*, *Process Step 2: Service Execution* and *Process Step 3: Ongoing Data Management and Maintenance* are applied to each transaction. Without focusing on the technical

[2]While the agent represents user interfaces, the proxy is invisibly placed between SPs and users. The agent carries out all tasks that require user interaction; the identity is mainly responsible for communications.

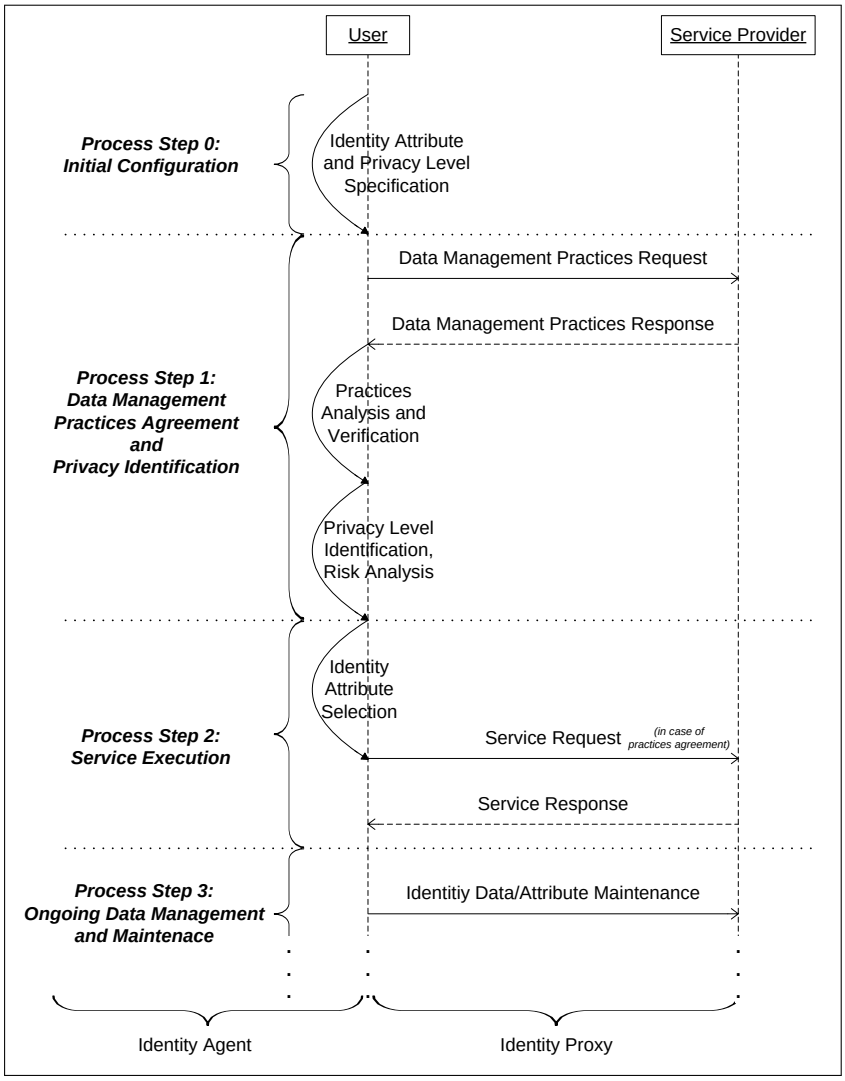



Figure 3.1: Simplified procedural framework method

implementation the next paragraphs specify those four steps. Short reviews of the user scenarios briefly apply the descriptions in real world examples. Even though a separation between identity agent and identity proxy was made, the common term *user tool* is from now on mostly used.

**Process Step 0: Initial Configuration**
In the first and initial process step users personalizes the user tools. They securely store their identity attributes in internal databases on the mobile phones. They also specify individual evaluations of different privacy degrees that are necessary for the following process step.

**Process Step 0: User Scenario Review** *Carol and Leon could have had prepared their mobile phones by storing identity attributes registration and ordering. They could also have had specified different levels of privacy. Carol for example could have had defined that using her data for marketing would result in a low privacy level. Leon, on the contrary, could have had specified that in case SPs request postal address, telephone number and personal interests at the same time, a low privacy level is in place, too. How to handle these levels is not part of the current process step.*

**Process Step 1: Data Management Practices Agreement and Privacy Identification**
This process step aims to agree on data management practices. The user tool thus automatically requests appropriate information from SPs and analyzes and verifies responses. Now, the preconfigured privacy evaluations help to identify whether users agree on SP specifications or not. If they disagree transactions are aborted; otherwise, the tool identifies the transactions' most probable privacy level and presents it on the screen. It also analyzes potential privacy risks and issues. Informed user consents then allow to progress with the next step.

**Process Step 1: User Scenario Review** *Carol and Leon could have had requested data management practices before exchanging identity attributes. Based on their personalized system configurations, the user tools then could have had verified whether Carol and Leon agree on the the specifications or not. If so, the tool could have had identified potential privacy levels and tried to analyze privacy risks and issues (e.g. user data concatenation) by reviewing transaction logs. The entire information could have had been presented on the phones so that Carol and Leon could have had been enabled to provide consents.*

**Process Step 2: Service Execution**
After data management practices, possible privacy levels and potential risks and issues are accepted, services are requested. To unburden users, the tool automatically selects requested attributes from the databases. Based on informed user consents those are integrated into pseudonyms and securely and anonymously sent to SPs. Then responses are interpreted and appropriate tasks carried out on the mobile phones.

**Process Step 2: User Scenario Review** *One main problem that led to privacy violation was that Carol felt bothered by large amounts of data required for registration. This could have been a reason why she processed the form while lacking concentration. In that situation the user tool could have had selected attributes from Carol's database that she specified in process step one. It then could have had asked to confirm and agree on the selection before securely and anonymously sending it as a pseudonym to the SP.*

**Process Step 3: Ongoing Data Management and Maintenance**
Privacy requires that transactions are not concluded after service responses are sent. It is rather very important to start managing and maintaining collected data at that point. Only by keeping collected identity attributes permanently updated, high privacy can be guaranteed. Thus, particular mechanisms like logging, reviews and ways to easily access and modify collected identity attributes on SP sites, are necessary.

**Process Step 3: User Scenario Review** *Granted that Carol did not register to only one but rather to many bulletin boards. She now received a new telephone number and wants to update her profiles. Visiting every single website would result in lots of work and time expenses. Here, the user tool could identify affected SPs that received this particular identity attribute and automatically present Carol with easy ways to modify this data at those sides. Furthermore, log files would allow reviews of past transactions (including disclosed identity attributes) and identifications of provided ways for access.*

#### Interim Conclusion

The overall procedural method is now specified in it's first draft, which is reviewed in this interim conclusion. It derives privacy and security protocol as far as user tool requirements and also defines the first functional requirement.

**FR 1. Guarantee coherent transactions and ongoing data management processes by providing privacy aware framework mechanisms.**

**Protocol Requirements** (discussed in Subsection 3.2.3)

- Allow request, verification and agreement on data management practices.
- Enable unique specification and automated selection of requested identity attributes.
- Ensure protection of anonymous data transfers.

**User Tool Requirements** (discussed in Subsection 3.2.4)

- Simple user interface to:
  - Anonymously browse the web.
  - Specify individual levels of privacy.
  - Enter and modify identity attributes in local databases and at SP sides.
  - Review and maintain pseudonyms of previous transactions.
  - Configure the user information flow related to system notification messages.
  - Review transaction logs.
- Functional mechanisms to:
  - Securely store privacy-sensitive identity attributes, pseudonyms and log files.
  - Analyze and interpret data management practices to identify potential privacy risks and issues.
  - Log and review transactions in such a way, that the information is accessible and readable both, by users and the user too.

### 3.2.3 Privacy and Security Protocols

In the previous subsection the procedural framework method was specified and related requirements derived. According to this, the next paragraphs discuss those privacy and security protocols that are necessary to support the method. Thus, by applying the first draft requirements, one or more protocols are necessary to:

1. Request, verify and agree on data management practices,
2. Uniquely specify and automatically select requested identity attributes and
3. Protect anonymous data transfers.

Some of those requirements may be covered by one and the same protocol. However, for easier understanding this devision is kept for now. Furthermore, extracts of the relevant process sequences from Figure 3.1 are presented, whereby respective steps are emphasized in italic and blue color.

### Request, Verification and Agreement on Data Management Practices

The first requirement is related to the beginning of process step one (see Figure 3.2). This protocol is responsible for allowing agreements between users and SPs on applied data management practices before any identity attribute is disclosed. From privacy aspects this protocol plays a major role in the proposed framework.

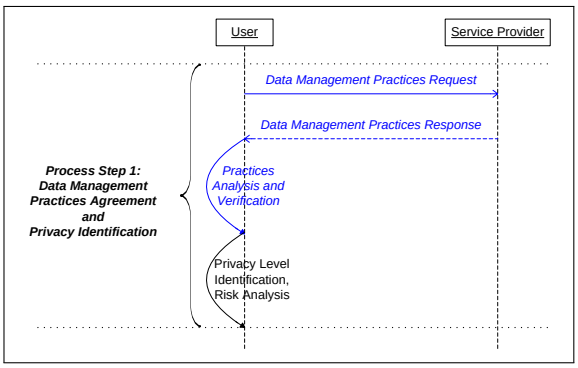



Figure 3.2: Protocol for request, verification and agreement on data management practices

In order to successfully agree on data management practices, corresponding policies are required to be published on SP sides, so that they can be automatically requested by the user tool. To be standardized, they need to be written in a language that is known to all participants. Those policies are required to provide information about the following privacy related aspects[3].

- Contact details of collecting company.
- Requested identity attributes.
- Purpose of data collection.
- Further companies that are able to access and process collected identity data.
- Storing periods for collected identity attributes.
- Accessibility of collected identity data.

Agreement on specified practices then requires users to hold one or more similar policies on their mobile phones. Those *user policies* are meant to represent data management practices, users allow SPs to apply[4]. In order to support simple comparison, those policies need to be written in the same vocabulary and format as SP ones. On this background, the user tool is required to match SP and user policies with the goal to identify potential agreements or disagreements. However, detailed consideration of the required mechanism is left out of the current specification.

[3] As it is described in Section 3.3, the already presented P3P approach (Subsection 2.2.5) covers most of the following specifications. However, this section here aims to describe protocol requirements for data management practice agreements rather than matching P3P to the intended solution.

[4] As a reason, that this section only defines requirements, no policy sample is provided.

### Unique Specification and Automated Selection of Requested Identity Attributes

The second protocol requirement aims to uniquely specify requested identity attributes (Figure 3.3). The ambition is, that the user tool automatically selects appropriate data from the local databases. Therefore, it is once again necessary that users and SPs work within the same vocabulary; a tight connection to the first protocol specification becomes visible.

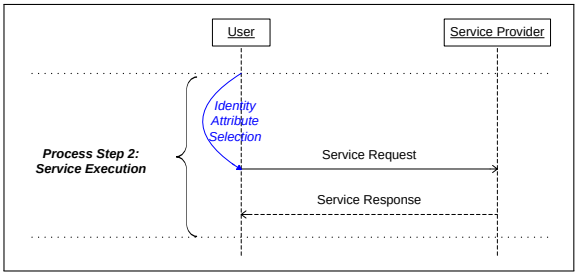



Figure 3.3: Protocol for unique specification and automated selection of identity attributes

According to Subsection 3.2.2, the procedural method requires users to run through an initial system configuration and to securely store identity attributes on the mobile phones. Granted now, SPs specify in their policies that *telephone number* and *birth date* are requested. If those attributes in the databases are stored within the same vocabulary as used in policies, the user tool is able to easily select them in an automated way. In cases requested attributes are not yet available, the tool is required to ask users to immediately provide them. The data then has to be stored under the same descriptions that SPs used in their requests.

The requirements analysis so far shows that the first two requirements can be covered by the same protocol. This is mostly due to the fact that the main aspect of both is to work within a common vocabulary.

### Protection of Anonymous Data Transfers

The last requirement of this subsection specifies a separate protocol that can not be integrated into the previous one. As Figure 3.4 shows, it is related to each process step in that information between users and SPs is exchanged. While protection on mobile phones is discussed in the next subsection, it is also very important to secure data just the same immediately after it left it's secure storage.

This protocol is responsible to protect and anonymize communications, starting with data management practice requests and *ending* with ongoing data maintenance. This prevents SPs from determining sending stations, that is to say users and their mobile phones. It guarantees high user anonymity and data security. The following nine protocol related specifications are defined.

- Encryption takes place both, in requests and responses in order to ensure that only users and intended SPs are able to read transmitted messages.
- User anonymity is guaranteed at any time.
- Encryption and corresponding key exchanges are based on open standards and protocols.

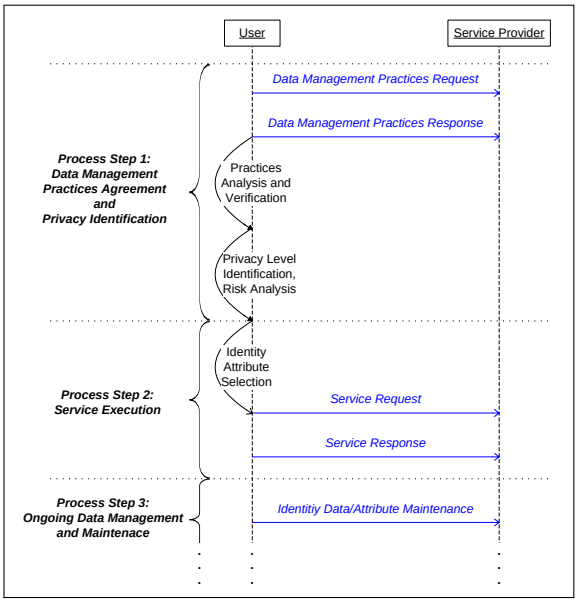



Figure 3.4: Protocol requirement for protection of anonymous data transfers

- Different encryption keys are applied in each transaction.
- Encryption is processed as fast as possible.
- Various levels of anonymity are available.
- Processing power is permanently kept at minimum; in particular during data encryption.
- Lowest amount of trust relations in other parties is required.
- Independence from underlying communication infrastructure is guaranteed; UMTS and WLAN are supported.

To enable anonymity of sending stations, a corresponding infrastructure is required. In this, all participants are prevented to read data messages and identify sources. However, for flexibility reasons participation should be voluntary. Thus, everybody needs to be free to send encrypted data packets also directly, non anonymously to SPs. Figure 3.5 visualizes these two different communication approaches.

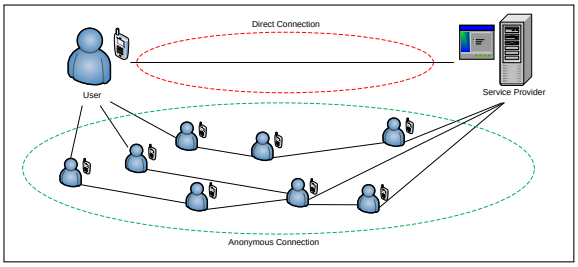



Figure 3.5: Direct and anonymous data communication

#### Interim Conclusion

The protocol requirements and specifications have shown that it is necessary to design two separate protocols. While the first one enables agreements on data management practices and automated selections of requested identity attributes, the second is responsible to protect anonymous data exchanges. In combination with the user tool presented in the next subsection and the underlying infrastructure, both protocols support the overall procedural framework method. Summing up Subsection 3.2.3, the following three functional requirements are adapted from the stated protocol requirements.

**FR 2. Allow agreements on data management practices.**

**FR 3. Support automated selections of requested identity attributes from the databases on mobile phones.**

**FR 4. Ensure secure and anonymous data communications.**

### 3.2.4 User Tool

The central part of the framework for IM and PM on mobile devices is a user tool. It integrates the procedural method and the privacy and security protocols into a user-centered system architecture. Running on mobile phones, it provides mechanisms to surf the web and to apply and administer identity data in privacy-sensitive ways.

The next paragraphs specify related requirements. They start with a presentation of the basic idea and the functional subsystems. Then they process with interface and mechanism related specifications and conclude with a definition of related functional requirements.

#### Basic Idea and Functional Subsystems

When designing and developing software one of the first steps is to specify the underlying platform. Thus, an extensive analysis came to the conclusion that Android[5] is best suitable. The four main decision criteria are that first, so far no identical functionality has been designed for this operating system. Second, it provides simple ways to integrate new third party applications. Third, the platform is free and open source and fourth, many investigations (i.a. [52] and [53]) predict that in near future more and more mobile phones designers will adopt Android. However, due to lack of time, no prototype has been developed. In some sense the specification of the Android platform seems to be irrelevant, however, theoretical feasibility studies for particular implementation approaches on this operating system were carried out whenever applicable.

Reviewing Figure 3.1, the user tool is responsible for the entire communication between users and SPs. For this, it provides the two functional subsystems *identity agent* and *identity proxy*, as stated in Subsection 3.2.2[6]. Figure 3.6 visualizes connections between subsystems, user tool and SPs. Thereby, communication is established through UMTS or WLAN channels.

[5] Android is an operating system based and the Linux kernel and built for mobile devices. More information can be found at the manufacture's website `http://www.android.com/`.

[6] The subsystems can generally be seen as user interface (identity agent) and functional mechanisms (identity proxy).

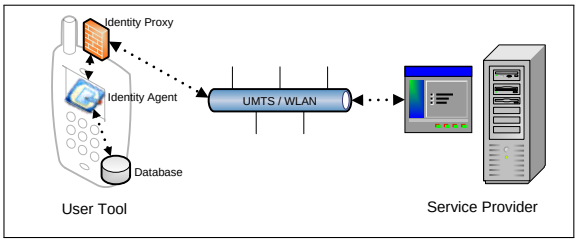



Figure 3.6: Functional system architecture

After the functional system architecture is specified, the first draft user tool requirements of Subsection 3.2.2 can be reworked. Therefore, the following list represents and extended version, whereby further aspects are added that originate from the previous investigations. For easier referencing purposes, all requirements related to the user interface are introduced as UIR1 to UIR8; requirements for user tool mechanisms as UTMR1 to UTMR7. While it is not necessary to further clarify the self-explanatory user interfaces at this stage, particular mechanisms on the other side still require additional specifications, as shown below.

Simple interactive user interface to:

- Anonymously browse the Web. (UIR1)
- Generate personalized user policies and evaluate related privacy levels. (UIR2)
- Enter and modify identity attributes in local databases and at SP sides. (UIR3)
- Review and maintain pseudonyms of previous transactions. (UIR4)
- Review transaction logs. (UIR5)
- Locate data management practices in human readable formats at SP sides[7]. (UIR6)
- Configure user information flows related to system notification messages[8]. (UIR7)
- Interact with the system; allow to accept or decline activities and enable consents. (UIR8)

Functional mechanisms to:

- Identify privacy levels for upcoming transactions. (UTMR1)
- Analyze and interpret data management practices to identify potential privacy risks and issues. (UTMR2)
- *Remember* and automatically apply login data. (UTMR3)
- Log and review transactions. (UTMR4)
- Securely store privacy-sensitive identity attributes, pseudonyms and log files. (UTMR5)
- Continuously inform users about current process steps and allow consents. (UTMR6)
- Enable immediate interrupts of data transfers, actions or entire transactions (provide an "Exit Button"). (UTMR7)

[7]This particular specification is required to be included in data management policies.
[8]The challenge is to balance information in sufficient detail and avoiding bothering by presenting too much notifications. A user-centered way to configure information frequency is required.

### Functional Mechanisms

The following paragraphs provide additional information about selected user tool mechanisms. All unmentioned aspects are either already discussed in sufficient detail or do not require any further clarification so far.

**Basic Mechanisms**
(UTMR3) One important advantage of IDMS infrastructures is the support of users to remember and automatically apply large amounts of account data. Without using such systems users tend to specify simplified passwords or the same for wide ranges of different services. Here, an appropriate mechanism is required that enables users with simply ways ti securely work with login data.

(UTMR4) Regarding logging of transactions it is important to save as much useful information as possible. This includes policy specifications, disclosure dates and transaction contexts. It is necessary that the user tool provides a consistent functionality that supports detailed reviews.

(UTMR5) One of the most critical mechanisms is to securely save privacy-sensitive data (identity attributes, pseudonyms and log files). The challenge here is to enable protected storages directly on mobile phones without requiring too much processing power. however, as mentioned in Chapter 1, encryption always comes with expenses. Thus, the balance between usability and security plays a major role. Furthermore, resulting from high risks in loosing mobile phones, security approaches are required to protect data sufficiently even in cases of loss.

**Particular Mechanisms**
The user tool requirements *Identification of personal privacy levels* and *Analysis of privacy risks and issues* (UTMR1 and UTMR2) belong to process step zero and one, as Figure 3.7 shows. They both can not be explained as briefly as the previous ones.

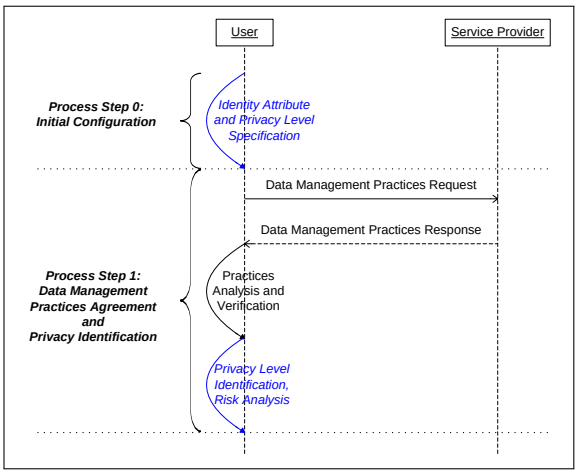



Figure 3.7: User tool requirement for identifying personal privacy levels and analyzing privacy risks and issues

According to the addressed policies that specify data management practices, they can not only be used for agreements between users and SPs. The included information rather provides valuable input to identify potential privacy levels and to analyze possible risks and issues. Therefore, the

same time users create individual policies on their mobile phones, they also need to specify and integrate personal evaluations of related privacies[9]. This enables the user tool to automatically identify and inform about potential privacy levels for upcoming transactions by carrying out the following process.

> Every time policies are received, the tool compares the information with every user policy. It analyzes match factors of user requirements that need to be fulfilled in order to reach defined privacy levels. After all policies are analyzed, the tool defines privacy levels by choosing the policy that indicates the highest match factor[10].

To provide users with even more precious information than just potential privacy levels, the tool needs to join current SP policies with information stored in logs. Thus, it can analyze possible privacy risks and issues that could occur in intended transactions. Granted the following situation:

**Privacy Risk Analysis** Leon previously ordered items from YouBuy. He was asked for postal address and telephone number, as accurately logged on Leon's mobile phone. YouBuy now requests the identity attributes birth date and identification number. The user tool running on Leon's mobile phone could detect this discrepancy and warn about possible user data concatenation risks.

To successfully implement such analyzing functionalities the system needs to *learn* identification rules. So, different privacy risks and issues associated with typical occurrences[11] are required to be specified and integrated during development processes.

Summing up these two mechanisms, small extensions to user policies are necessary. To enable analyses of privacy levels, users are required to integrate personalized evaluations. It is also important that rules for analyzing privacy risks and issues are directly integrated into the system, because those decisions are outside the users' knowledge and business.

### Interim Conclusion

Concluding the protocol specification, the following two functional requirements are defined. They strongly interact with the presented user interface requirements (UIR) and the requirements for the user tool mechanism (UTMR).

**FR 5. Provide simple interactive user interfaces that fulfill all UIRs.**

**FR 6. Provide functional mechanisms that meet all UTMRs.**

[9] Samples: A SP collects postal information and indicates to forward this data to marketing companies. This behavior results in a low privacy level. Another company that only requests username and password could be seen as working on a high level of privacy. To support unexperienced users best possible a reference list is required that contains sample transactions with suggestions of related privacy levels.

[10] The framework assumes that presenting privacy levels leads to first user impressions or can be seen as a decision criterion whether to disclose requested identity attributes or not.

[11] Comparable to the sample presented in the gray colored box.

### 3.2.5 Conclusion

Section 3.2 presented the requirement and specification analysis. A discussion of the framework parts procedural method, privacy and security protocols as well as user tool worked out the key specifications. Furthermore, a quick outlook into the underlying infrastructure and applied technology emphasized on the treated context.

Referring again to Figure 1.1, the design preparation is now completed. Thus, the following Section 3.3 describes the subsequent development process. Therefore, Table 3.2 summarizes the six functional requirements that need to be met by the proposed framework solution.

Table 3.2: Functional framework requirements

| **Framework Part** | **Functional Requirement** |
|---|---|
| Procedural Method | FR 1. Guarantee coherent transactions and ongoing data management processes by providing privacy aware framework mechanisms. |
| Privacy and Security Protocols | FR 2. Allow agreements on data management practices.<br>FR 3. Support automated selections of requested identity attributes from the databases on mobile phones.<br>FR 4. Ensure secure and anonymous data communications. |
| User Tool | FR 5. Provide simple interactive user interfaces that fulfill all UIRs.<br>FR 6. Provide functional mechanisms that meet all UTMRs. |

## 3.3 Design and Development

Section 3.3 presents the design of the framework according to the information of the previous section. It specifically aims to meet the stated functional requirements and the Privacy Design Requirements. This section generally adopts the overall structure of Section 3.2. However, it is necessary to make up the design of one protocol in front, because without this preparation it is not possible to specify the final version of the procedural method. After this, all framework parts are designed and finally concluded by a short summary.

### 3.3.1 Design Preparation

It is necessary to perform the design of the protocol to request, verify and agree on data management practices prior to the procedural method extension. Thereby, the following paragraphs work in tight connections to the privacy technology architecture design approaches of Subsection 2.2.5.

### General Foundations

Section 3.2 briefly mentioned that the P3P approach, as introduced in Subsection 2.2.5, seems to be best suitable to support agreements on data management practices in the project context. However, it is all the more remarkable that so far just a few IDMS solutions took P3P into consideration. Three of them are also cited in the Literature Review, namely [6], [12] and [16]. The fact, that the first system works on a different basic idea than this project, makes it only usable with reference character[12].

On the contrary, the second and third project are helpful in the sense that they provide an extension to the P3P vocabulary that supports position based data of roaming users[13]. This extension, slightly modified as shown in Table 3.3, is adapted to the framework solution. By adding time and position (Global Positioning System (GPS) coordinates) based categories, the support of location based applications is continued and at the same time anonymity of sending stations guaranteed.

Table 3.3: P3P data schema extension

| Type | Allowed Categories | Allowed Descendants | Short Description | Notes |
|---|---|---|---|---|
| location | Locational and Time-Based Data | time, position | Position based device information | Used to request and provide information required for locational based services. |
| time | Locational and Time-Based Data | ymd.year, ymd.month, ymd.day, hms.hour, hms.minute, hms.second | Current device time and date | Used to request and provide information required for locational based services. |
| position | Locational and Time-Based Data | | Current device position (GPS coordinates) | Used to request and provide information required for locational based services. |

If SPs want to include this P3P extension, particular statements need to be added to their policies. Even though, the content of those files is shown later, Listing 3.1 anticipates and briefly shows the addressed definitions in an eXtensible Markup Language (XML) syntax.

```
1 <STATEMENT>
2   <DATA-GROUP>
3     <EXTENSION>
4       <DATA ref="#location.time.ymd.year"/>
5       <DATA ref="#location.time.ymd.month"/>
6       ...
7       <DATA ref="#location.time.hms.second"/>
8       <DATA ref="#location.position"/>
9     </EXTENSION>
10  </DATA-GROUP>
11 </STATEMENT>
```

Listing 3.1: P3P policy extension

[12]Instead of aggregating identity attributes into pseudonyms for current transactions, it enables users to appear to SPs based on different preconfigured pseudonyms.

[13]The default P3P vocabulary is listed in the Appendices A.3.1 and A.3.2.

### Implementation Overview and P3P Data Elements

SPs that want to work privacy-sensitive in the framework context are required to create and publish two P3P data elements;a *policy reference file* and a *policy*. Both are written in XML format that complies with P3P specifications and vocabulary. As a reason that creation of these files is complicated for beginners, a couple of different editors have been designed. IBM's *P3P Policy Editor*[14] is one of them. It's use is for free but the provided interface is not that simple and intuitive. Much more comfortable and easier to handle is the *P3P Wizard*[15]. It leads users interactively through comprehensible dialogs and provides the files as email attachments afterwards. However, this online service is not free of charge.

**Policy Reference File (P3P_Reference.xml)** Contains location of corresponding policy. Specifies a local Uniform Resource Identifier (URI) or set of URIs that are covered by the policy.

**P3P Policy (P3P_Policy.xml)** Main element of P3P. Specifies data management practices that are applied to URIs, as listed in the related reference file.

The following Listing 3.2 presents the content of a sample policy reference file. Listing 3.3 then shows a part of the P3P policy itself; it applies the P3P vocabulary of Appendices A.3.1 and A.3.2. The entire policy, including explaining comments, is placed in Appendix A.3.3. That appendix also shows a step by step guide to simplify creation of P3P policies.

The reference file defines these three conditions in XML P3P syntax:

- All statements are valid for one day (86400 seconds); thereafter the file needs to be renewed.
- The corresponding P3P policy is located at *./P3P/P3P_Policy.xml*.
- P3P policy specifications are applied to all resources whose path begins with */register*.

```
1 <META xmlns="http://registration.example.com/">
2   <POLICY-REFERENCES>
3     <EXPIRY max-age="86400"/>
4       <POLICY-REF about="/P3P/P3P_Policy.xml">
5         <INCLUDE>/register/*</INCLUDE>
6       </POLICY-REF>
7   </POLICY-REFERENCES>
8 </META>
```

Listing 3.2: Sample P3P policy reference file

The policy file makes these five statements in XML P3P syntax:

- YouBuy is the collecting company.
- A privacy policy in human readable format is available (disc.html).
- Access to collected online and physical contact information is allowed.
- The company's customer service is available to resolve disputes related to privacy violations.

[14]The P3P Policy Editor is available at `http://www.alphaworks.ibm.com/tech/p3peditor`.
[15]The P3P Wizard is available at `http://www.p3pwiz.com/`.

- Given and family name are the requested identity attributes. They are 1.) used for the current transaction, marketing and telemarketing, 2.) provided to third parties with divergent privacy practices (opt-out possibility) and 3.) stored according to laws and legal requirements.

```
<POLICIES xmlns="http://www.w3.org/2002/01/P3Pv1">
<POLICY name="youbuy_order"
    discuri="http://www.YouBuy.com/P3P/disc.html" <!-- Human redable policy -->
     xml:lang="en">
<ENTITY>
  <DATA-GROUP>
    <DATA ref="#business.name">YouBuy</DATA>
    <DATA ref="#business.contact-info.postal.street">Grooseveien 36</DATA>
    <DATA ref="#business.contact-info.postal.city">Grimstad</DATA>
  </DATA-GROUP>
</ENTITY>
<ACCESS><ident-contact/></ACCESS>
<DISPUTES-GROUP>
  <DISPUTES resolution-type="service" service="http://www.YouBuy.com/contact.asp"
      >
  </DISPUTES>
</DISPUTES-GROUP>
<STATEMENT>
  <CONSEQUENCE> For delivery reasons, customer contact details are collected. </
      CONSEQUENCE>
  <PURPOSE> <current/><contact/><telemarketing/> </PURPOSE>
  <RECIPIENT> <ours/> <other-recipient  required="opt-out"/> </RECIPIENT>
  <RETENTION> <legal-requirement/> </RETENTION>
  <DATA-GROUP>
    <DATA ref="#user.name.given"/>
    <DATA ref="#user.name.family"/>
  </DATA-GROUP>
</STATEMENT>
</POLICY>
</POLICIES>
```

Listing 3.3: Sample P3P policy file

### Framework Process Sequence

In the framework context every policy request is carried out according to the same scheme. Users browse the internet with the P3P enabled user tool. The tool continuously checks whether a received Hypertext Transfer Protocol (HTTP) response header includes a particular *policyref* data element or not[16]. If so, it does not display any content of the response and instead initiates a HTTP GET request[17] to ask for the specified reference file. In the next step the user tool analyzes the received information in order to fetch the corresponding P3P policy at the stated location. It then analyzes and compares the information to the user specifications (more information about this is given in the next two subsections).

[16] This data element defines that identity attributes are requested. Thus it indicates the reference file that provides the location for the applied P3P policy.

[17] A GET request is one of many HTTP methods. It aims to retrieve information that is identified by the URI specified in the request.

Figure 3.8 shows a UML sequence diagram, based on [54], that visualizes the described process steps in order to receive SP policies. However, while in first place it looks very complicated for users, all steps are automatically carried out by the P3P enabled user tool in the background. In addition to this figure, Listings 3.4 and 3.5, inspired by [55], briefly show data messages of typical requests and responses. Here, the response includes a P3P header that indicates the location of the corresponding reference file.

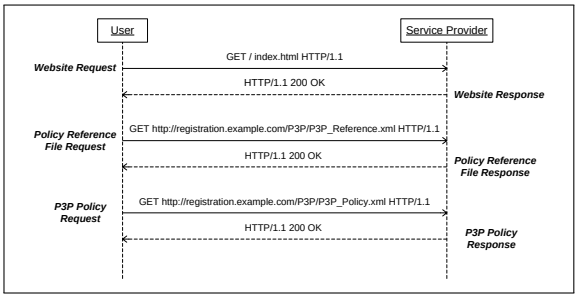



Figure 3.8: Process Sequence for requesting and receiving P3P policies

```
1 GET /index.html HTTP/1.1
2 Host: registration.YouBuy.com
3 User-Agent: Mozilla/4.0
4 Accept: */*
5 Accept-Language: */*
6 Connection: close
```

Listing 3.4: GET request

```
1 HTTP/1.1 200 OK
2 P3P: policyref="http://registration.
      YouBuy.com/P3P/P3P_Reference.xml"
3 Content-Type: text/html
4 Content-Length: 7413
5 Server: CC-Galaxy/1.3.18
6 ...content follows...
```

Listing 3.5: HTTP response

Briefly addressing the approach of comparing user and SP policies, the common XML syntax eases this task. It enables searches for the nine P3P tags[18] and the *discuri* specification[19]. In this way, agreements between users and SPs can be identified. However, extensive discussion about comparing P3P policies is out of the current scope and rather presented in Subsection 3.3.4.

#### Interim Conclusion

As shown, this protocol can almost entirely be realized with P3P technology. There are only two slight modifications necessary. First, the applied vocabulary is extended in order to cover mobility aspects. Second, users are responsible to create their own, personal policies. In which way to support users in creating user policies, how to compare those and what to do with the results is discussed in Subsection 3.3.4.

### 3.3.2 Procedural Method

Now that the very important protocol to agree on data management practices is designed, the procedural method can be finalized. To avoid redundancy, specifications that are not modified, like

[18] Entity, Access, Disputes, Purpose, Recipient, Retention, Data-Group, Statement and Categories (Appendix A.3.1).
[19] The *discuri* element specifies existence and location of the human readable policy. See Listing 3.3, line 3.

for example the storage of identity attributes in process step zero, are not repeated. Focus is rather put on those processes that need restructuring.

### Adjusted Basic Idea and Detailed Procedure

SP are required to create P3P policies for each website content that requests identity attributes. Thereby, one policy is able to cover more than a single URI (the mapping needs to be specified in the reference file). On the other hand, users are required to specify one or more personalized user policies that reflect their own demands and are stored on the mobile phones. Furthermore, users are required to evaluate potential privacy levels for each policy and directly integrate this into the policies.

After this preparation, users are able to browse the web with the user tool. Particular framework configurations on SP sides then lead to the result, that as soon as a website requires user identification, a modified HTTP header is responded. Thereby, this header specifies the location of the corresponding policy reference file[20]. The existence of this particular data element is automatically recognized by the user tool. It blocks to display corresponding input forms because requested data is directly sent to SPs rather than filled into any website. Instead, the user tool automatically requests the specified P3P reference file and policy.

It then analyzes and verifies the applied practices by matching SP specifications to user demands. In this way it is able to identify potential privacy levels. In addition, the tool also extracts information about human readable policies and displays appropriate information on the screen. If users agree on the so far analyzed results, a privacy risk analysis is carried out by comparing current requests with previous transactions stored in log files.

In the next step the tool automatically selects requested identity attributes from the local databases and stores them in data files that represent pseudonyms for current transactions. In order to work anonymous, as described in Subsection 3.3.3, pseudonyms are then divided into several data packets, encrypted and sent through an anonymous network infrastructure. Thereby, the destination address is the same as specified for the *submit* button of the blocked form.

In one of the last steps SPs that integrate a particular framework protocol decrypt and merge received data packets and interpret the request. In case one or more packets are corrupt or missing, they are rerequested. Otherwise SPs transform the message into a POST request[21] because it is used as input for the intended online form[22].

Finally, SPs use the same anonymous network to reply with their responses. The user tool then analyzes and displays corresponding information on the mobile phone screens. Thereby it is important, that all critical processes are only carried out with informed user consents in order to meet the privacy requirements of Table 3.1.

[20] In the user scenarios this would be the case when Carol opens the registration page and Leon clicks the checkout button.

[21] The HTTP POST method is mostly used in combination with forms. It helps to send formula data to servers.

[22] Without applying the user tool users would have directly sent data as such POST requests by filling out online forms.

Independent of the previous process steps, the user tool provides an easy way to update identity data. Therefore, users modify attributes in their database, whereby changes are automatically recognized by the user tool. It identifies affected SPs and displays corresponding information on the mobile phone screens that ease updates on SP sides.

The UML sequence diagram in Figure 3.9 visualizes the previously described final modification and extension of the procedural method (a higher resolution is placed in Appendix A.1). Thereby, the entire communication is encrypted by the identity proxy (represented in the middle box).

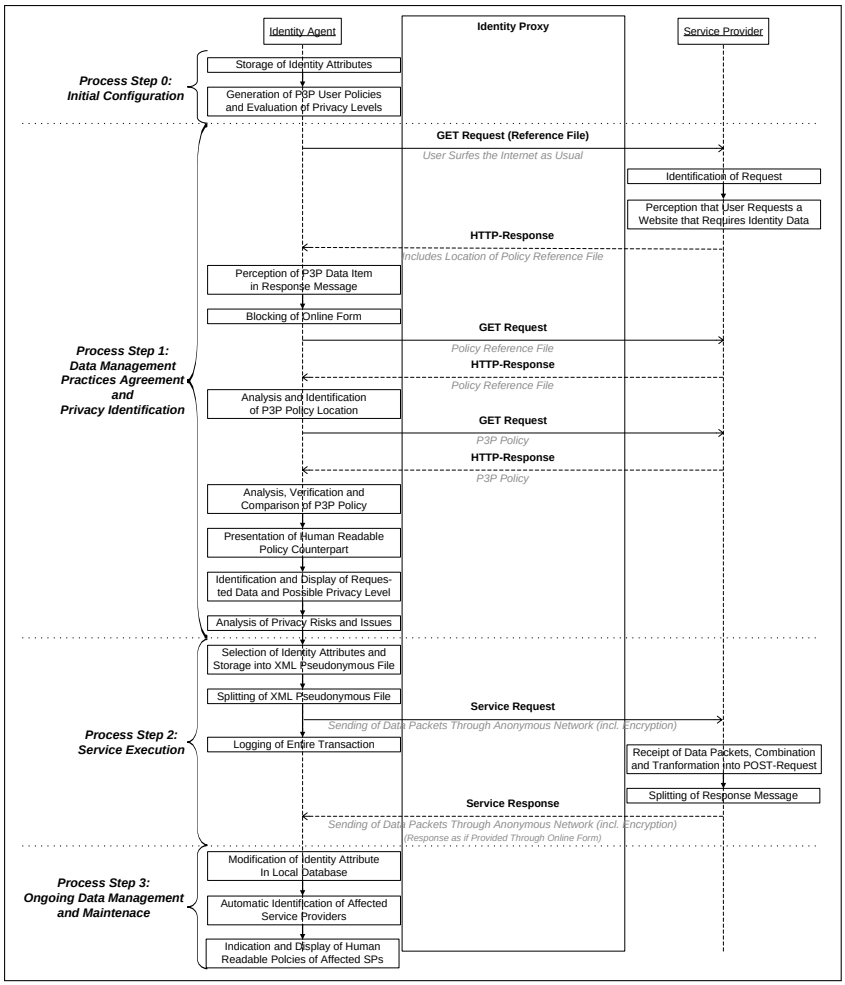



Figure 3.9: Finalized procedural framework method

## Interim Conclusion

This subsection finalized the procedural framework method by introducing further mechanisms and processes. The following two subsections design those framework parts that are necessary to support the method. As a conclusion, it is important that in cases where SPs do not make P3P policies available, the user tool is still able to process transactions in an anonymous way. However, it

then displays the default website content together with an appropriate warning that the intended transaction is not treated privacy-sensitive and that no user support can be provided.

### 3.3.3 Privacy and Security Protocols

With reference to the requirements analysis of Section 3.2, three protocol specifications are defined:

- Allow request, verification and agreement on data management practices.
- Enable unique specification and automated selection of requested identity attributes.
- Ensure protection of anonymous data transfers.

even though the first protocol was already designed in the beginning of this section, it is for consistency shortly reviewed in the next paragraph. The other two specifications are then handled in the following discussions.

#### Request, Verification and Agreement on Data Management Practices

In order to modify and extend the proposed framework method, the protocol for agreement on data management practices was already discussed in Subsection 3.3.1. So, to avoid redundancy, only the most important facts are summarized in Table 3.4.

Table 3.4: Review of the protocol for agreement on data management practices

| Actor | Activity |
|---|---|
| **General** | Adaptation and slightly modification of the P3P functionality.<br>Automated identification of reference file locations by user tool.<br>If reference file locations included, blocking of website content and automated request of corresponding P3P policy by user tool.<br>Comparison of user policies with SP ones by user tool in order to identify potential privacy levels.<br>User notification of human readable policies by user tool. |
| **Users** | Specification of one ore more policies that reflect individual, personal demands.<br>Evaluation and integration of expected privacy levels into each policy.<br>Storage of policies on mobile phones. |
| **Service Providers** | Modification of HTTP response headers on the web servers.<br>Definition of one or more reference files that indicate the policies' validity for particular URIs.<br>Specification of one or more policies for websites that request identity attributes.<br>Publication of reference files and policies at specified locations.<br>Publication of human readable policies at specified locations. |

As an addition, SPs need to specify reference file locations in their HTTP headers by integrating a particular *policyref* data element, as shown in Subsection 3.3.1. Those modified headers can then be connected to websites that request identity attributes, so that querying clients automatically receive information about corresponding locations. However, this is rather a SP requirement than a framework component. Therefore, further technical instructions are not provided and it is rather referred to the tutorials at [55].

### Unique Specification and Automated Selection of Requested Identity Attributes

As worked out, the key success factor for unique specification and automated selection of identity attributes is a common language between users and SPs. Therefore, an application of the P3P vocabulary that is used in privacy policies meets this requirement in an easy way.

**Foundations and Database Structure**
By reviewing Appendix A.3.2 it seems that there are a lot of different P3P data types to deal with when storing and automatically selecting attributes from local databases. However, having a more precise look the types *third-party* and *business* are out of the users' scope and rather used by SPs to specify collecting companies and affected third parties. Furthermore, all data elements belonging to the *dynamic* type do not have fixed values. Thus, even they can not be stored in databases.

As a result, identity attributes belonging to the *user* data type are the only one to store on the mobile phones. Thus, they are the once to consider during the current protocol design. The P3P data schema of Appendix A.3.2 shows that the *user* data type has seven descendants[23]. These are *name*, *bdate*, *login*, *cert*, *gender*, *home-info* and *business-info*. Based on this information the following database structure is proposed:

> The first five descendants specify a maximum of fourteen different data elements (e.g. *name.prefix*, *name.given* or *name.family*). This is a number that can easily be integrated into one single database table, named *user*. For efficiency reasons the other two descendants (*home-info* and *business-info*) are extracted and stored in two separate tables, named *home-info* and *business-info*. All three tables are then interconnected with so-called primary keys[24].

Figure 3.10 visualizes this database structure by showing samples of the three addressed tables, whereby the *ID* data field represents the primary key. The user tool is responsible to comply with this structure whenever it stores identity attributes.

[23] A descendant is a data type or an element that subdivides it's parent. For example *name* can specify a *user* in order to achieve the data element *user.name*.

[24] This project assumes that mobile phones are only used by single persons. So, each database contains identity attributes of one user. Furthermore, tables in the current framework version store one user dataset per person. So, on closer inspection primary keys for interconnecting seem to be redundant. However, in order to enable potential extensions and to assure well structured databases they are still used.

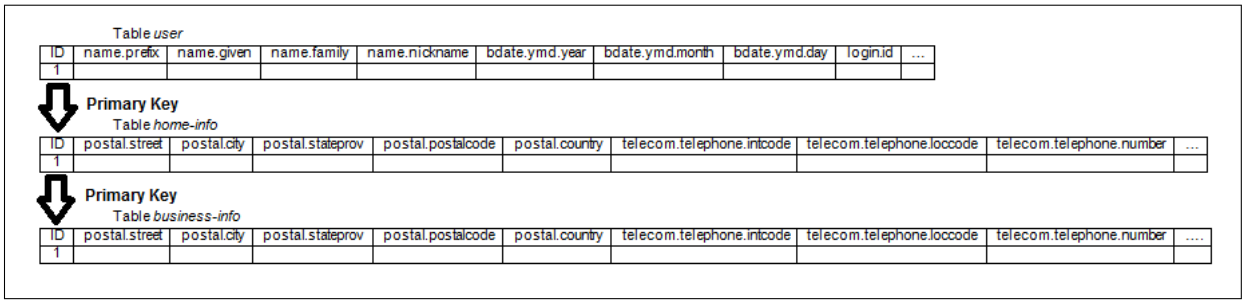



Figure 3.10: Database structure for identity attributes

### Selection Process

According to the designed database structure, selections of identity attributes are processed in the following way.

> The user tool analyzes SP policies and tries to locate the <DATA GROUP> in the <STATEMENT> XML tag (see e.g. line 22 of Listing 3.3). This is the part that includes information about requested identity attributes. The tool then analyzes the data group line by line. Granted, that one line specifies the identity attribute *user.home-info.postal.street*, the first part of the string (*user*) indicates that the desired attribute is most likely be stored in the database. The tool also knows that *home-info* specifies one of the three tables and thus queries it for the identity attribute that is stored as *postal.street*.

In case SPs request attributes that are not existent in the databases so far, the user tool displays according notifications on the screen. Users then are able to provide particular attributes that subsequently are automatically added to the databases, within the same vocabulary as used in the SP request.

After each line of the <DATA GROUP> tag is processed, selected identity attributes are integrated into XML files that represent pseudonyms for current transactions. Their format looks familiar to the one of P3P policies. The intention behind this is that the P3P namespace can be reused. One part of such a sample XML file that represents Leon's pseudonym used in a transaction with YouBuy, is outlined in Listing 3.6.

```
1 <STATEMENT xmlns="http://www.w3.org/2002/01/P3Pv1">
2     <DATA-GROUP>
3         <DATA ref="\#user.name.given"> Leon </DATA>
4         <DATA ref="\#user.name.family"> Morrison </DATA>
5         <DATA ref="\#user.bdate.ymd.year"> 1980 </DATA>
6         <DATA ref="\#user.bdate.ymd.month"> 08 </DATA>
7         <DATA ref="\#user.gender"> ...
8   </DATA-GROUP>
9 </STATEMENT>
```

Listing 3.6: A part of Leon's pseudonym applied at YouBuy's online shop

It can be the case, that users request services from SPs that they already disclosed identity attributes to. To cover those situations efficiently, the user tool automatically analyzes log files before processing the aforementioned task. Thereby, it tries to analyze the following aspects:

- Did the user already disclose identity attributes to the currently addressed SP?
- If so, were those attributes the same than currently requested?
- If so, were they disclosed to the currently queried URI?

If all questions can positively be answered, the user tool tries to locate a previously disclosed pseudonym. Finally, when this search is successfully, it skips the selection process and directly reuses the identified pseudonym.

**Pseudonym Files**
As it is shown in a later step, a user interface allows to easily review disclosed pseudonyms and identity attributes. For this, it is necessary to store pseudonyms in a consistent syntax. Therefore, *YYYYMMDD_SPN_LX.xml* is worked out to be the most proper one. The following Table 3.5 clarifies the meanings of the used abbreviations.

Table 3.5: Syntax of pseudonym file names

| **Abbreviation** | **Meaning** |
|---|---|
| YYYYMMDD | Creation date. |
| SPN | First three letters of requesting SP. |
| LX | Potential privacy level that was identified by user tool. |
| 20100611_YOU_L7 | Pseudonym with *privacy level 7* that was sent to *YouBuy* on *11.06.2010*. |

To overcome the challenge that several company names can consist of the same first three letters, a particular mechanism is designed. It applies consecutive numbers to the file names and starts with *02* for the first affected SP (e.g. YOU02 for a company that is called *YouTown*). Corresponding details are then logged in the databases in order to allow unique mappings.

**Brief Recap**
The P3P vocabulary provides great benefits for the protocol that uniquely specifies and automatically selects requested identity attributes. Due to the firstly designed protocol for agreements on data management practices, it is possible to reuse the suggested implementation without the need to develop a separate solution.

If the proposed database structure that complies with the P3P vocabulary is applied, the only necessity is to interpret SP policies accordingly. Selected attributes are then integrated into XML pseudonyms that are securely and anonymously sent to SPs. The corresponding protocol for this process is designed in the upcoming paragraphs.

### Protection of Anonymous Data Transfers

The next paragraphs design the protocol to protect anonymous data transfers. By addressing two reference systems they also show the context this work fits in. The main challenge of this protocol is not related to security but rather lies in the design of an appropriate infrastructure. Therefore, these paragraphs work out a suitable solution that can be used as communication platform for all framework communications between users and SPs.

**Reference Systems**
The upcoming protocol design is inspired by the TOR project[25]. In that solution each station is equipped with corresponding client software, whereby most stations act as directory servers that hold lists to specify other participants in the network. With the aim to react to network changes, these lists are updated on regularly basis. However, those updates require significant processing power and battery consumption. This aspect and the fact that TOR requires many trust relations in central servers leads to the conclusion that the approach is not entirely applicable in this project.

A second reference system is a solution proposed by Ardagna et al. in [34]. The general idea of their anonymous network provides a good basis, however, it fails to guarantee perfect user anonymity. In particular, they apply cryptography based on fixed keys that are exchanged between users and SPs in advance. While communication links are then kept anonymous, SPs still know sending stations.

As a conclusion, the developed protocol is required to overcome the addressed design flaws and provide a more valuable solution by guaranteeing entire user anonymity rather than just anonymous communication links. It is also supposed to consume less processing power. Under these conditions Subsection 3.2.3 defined nine protocol related requirements that need to be considered.

1. Encryption in both communication directions.
2. Continuous and entire user anonymity.
3. Encryption and corresponding key exchange based on open standards.
4. Different encryption keys in each transaction.
5. Encryption as fast as possible.
6. Availability of various anonymity levels.
7. Processing power as low as possible.
8. As lowest trust relations in other parties as possible.
9. Independence from underlying communication infrastructure.

**Anonymous Network Infrastructure**
The basic idea of this protocol is to use an artificial network that consists of all mobile phones with an installation of the proposed user tool (see Subsection 3.3.4). In this network each phone acts as an interstation on the data's way to SPs. Thereby, an appropriate protocol on the mobile phones enables to interpret and process received data packets accordingly.

In case users want to send data to SPs, they divide the original message into different, smaller packets, whereby this project defines the default amount of five. This value is chosen because it reflects a good balance between anonymity and transmission speed; higher numbers increase anonymity but also cause higher transaction times. However, users are allowed to adjust the default configuration in every transaction. Thus, in case they deal with identity attributes worth to be especially protected, they are free to increase the number, so that different levels of anonymity are supported.

[25] "TOR is a free software and an open network that helps you defend against a form of network surveillance that threatens personal freedom and privacy, confidential business activities and relationships, and state security known as traffic analysis. TOR protects you by bouncing your communications around a distributed network of relays run by volunteers all around the world." For more information it is referred to the official website at `http://www.torproject.org/`.

After division, each packet is sent to a different interstation that either forwards it to another interstation or sends it directly to the SP. In order to face the anonymity aspect that interstations could identify whether they received packets from sending mobile phones or just another interstation, a particular decision criterion is integrated into each packet.

Corresponding to this, [34] proposed that interstation act on a 50:50 probability to either forward or deliver packets. However, this approach is not applicable here because processing power at interstations needs to be kept as little as possible. Thus, with reference to the amount of data packets a random number (defined as *hop count*) is integrated. By definition, it's pool is zero to $<$*amount of packets*$>$. Hop counts then specify how many interstations packets are required to pass, before they are finally delivered. Therefore, interstations that receive packets locate this hop count element. If it is higher than zero, they decrease it by one and forward the packet to another station. In case the count is zero, they directly deliver the packet. This approach makes it impossible to identify sending mobile phones.

Figure 3.11 visualizes the *hop count* process sequence. This simplified figure shows that the user tool wants to send a single data packet with an integrated hop count of two. That indicates, that the packet needs to pass three interstations before it is finally delivered[26].

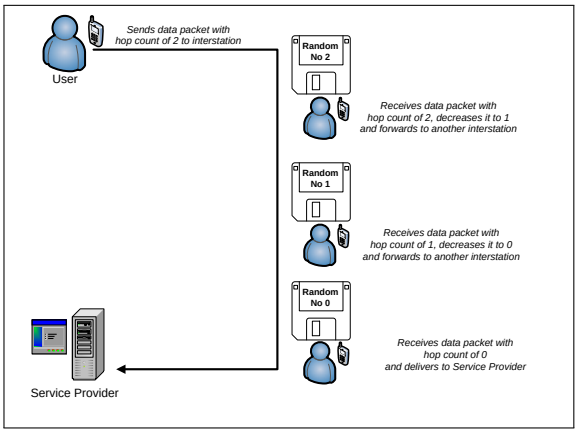



Figure 3.11: Packet forwarding based on an integrated hop count

**Process Sequence and Data Packet Design**

To use one and the same communication path for requests and responses interstations are required to log transaction details. In particular they need to remember the stations they received packets from and the ones they forwarded them to. Interstations also log so-called *message identifiers*. These identifiers are created in a predefined format to designate related packets. Identifiers for packets belonging to requests end with a *q*; response packets instead with a *r*. This specification, together with the temporary logged information[27] enables interstations to forward responses in reversed order by identifying corresponding packets in the logs and reviewing appropriate transaction details.

[26] Figure 3.11 does not show the entire process of anonymously sending data but rather describes how hop counts are handled.

[27] For privacy reasons, data is only logged temporary, so that after interstations processed packets belonging to a particular response, logs are automatically deleted.

Almost the same technique is applied at SP sites where the identifiers enable to determine packets that belong to one and the same request message. The numbers tell SPs, into how many packets their responses need to be divided before sending. SPs then integrate the same identifiers that were included in the requests, but this time they attach a $r$. Therefore, it is sufficient if SPs only remember message identifiers and interstations they received packets from. Based on this information they then send packets to the same interstation they received the belonging requests from. In the final step, the mobile phone that initiated the request, receives response packets and joins them in order to interpret the intended message.

The up to here described process sequence requires a particular data packet design, as shown in Figure 3.12. The general structure for request and response messages is the same. Furthermore, each data packet consists of two parts, whereby the first one (yellow color) represents the header that is transmitted in plaintext and the second one (blue color) includes the privacy-sensitive message that is encrypted, according to the approach described shortly.

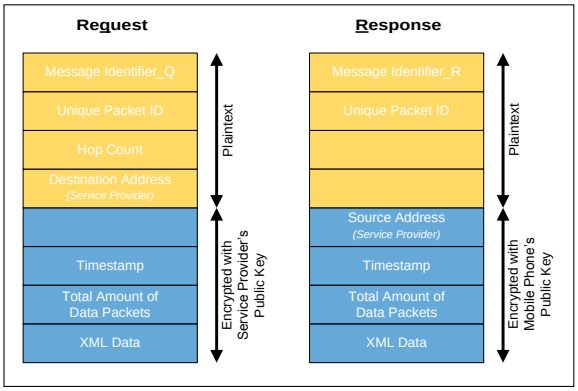



Figure 3.12: Data packet design for anonymous transmissions

The plaintext consists of the following data fields:

- **Message Identifier** - Specifies the message that the data packet belongs to. Identifier of requests and responses are the same but differ in their endings.
- **Unique Packet ID** - Indicates the particular packet.
- **Hop Count** - Defines how many interstations the data packet is required to pass[28].
- **Destination Address** - Specifies the target of the data packet; here the SP[29].

The encrypted part includes the following data fields:

- **Source Address** - Specifies the source of the data packet[30].
- **Time Stamp** - Defines the point of time when the data packet was sent.
- **Total Amount of Data Packets** - Specifies the total amount of data packets that belong to the overall message; this allows to carry out completeness checks.

[28]The *Hop Count* is only specified in requests because paths for response messages are fixed.
[29]The *Destination Address* is only specified in requests; it is not required for responses because interstations log appropriate data. Besides this, the sending mobile phone is anonymous, anyway.
[30]The *Source Address* is only specified in responses; for reasons of anonymity it is left blank in requests.

- **XML Data** - Represents pseudonyms in requests and confirmation messages or other data in responses.

**Anonymous Addressing Scheme**
To allow the sending of data packets, the appropriate contact information of participating interstations is required. However, one key feature of the designed anonymous network is that participants do not know about each other. Thus, to ensure anonymity on a very high level, they do not hold any contact information, in contrast to the TOR approach; each participant is rather registered at a central unit.

This unit holds addresses of all mobile phones in the entire anonymous network. When a user tool or an interstation wants to send data packets, it is required to request contact information of other interstations from this unit. Moreover, because of privacy reasons, received contact details are deleted from the phones after transactions have been completed are aborted.

The main challenge in applying this central station approach lies in the contact information itself. Simply using IP addresses could be one solution; however, mobile phones in communication networks like UMTS or WLAN receive dynamic addresses that change with each dial-up[31]. Thus, it is possible that the registration unit provides expired contact details what makes IP addresses unsuitable for this approach.

In fact, to achieve an unique and permanent addressing scheme, static IP addresses are required. For this purpose there are two possibilities; one is to request (mainly buy) such addresses from Network Service Providers or to use appropriate third party solutions (e.g. the service *fixed.IP*[32]). The other is to apply Domain Name System (DNS) services that ensure consistent addresses over time, like for example *DynDNS*[33]. DynDNS is able to automatically receive and process IP address changes, so that the registration unit is able to provide DNS names of interstations as contact information that is valid (almost) any time.

It is also important that even though the same anonymous infrastructure is used for responses, SPs are not required to ever contact the registration unit. Thus, the reuse of applied communication paths, as discussed previously, contributes to a high level of anonymity. Before finally the required encryption is designed, Figure 3.13 visualizes the anonymous network architecture that is discussed so far.

**Data Encryption**
For the coverage of security aspects there are a couple of different cryptography concepts available. Thereby, this work assumes session keys to be most suitable for the designed anonymous network. The decision is mainly based on two arguments. First, those keys enable sending stations to continuously stay anonymous because users suggest keys through the anonymous network that are then used in requests as well as responses. Moreover, they are changed for every transaction. Second, in contrast to asymmetric keys they provide much faster computations while requiring less processing power and battery consumption; and especially these components are limited on mobile phones.

[31] Following, basic knowledge about IP addresses is helpful. In case this is missing, Appendix A.5 gives a quick primer into that working area.
[32] The manufacture's website is located at `http://www.mdex.de/start/produkte/mdex-fixedip/`.
[33] DynDNS is just one of many identical services that enjoys great popularity and is free to use. Because of this, it is the primary choice for this project. For more information see the manufactures website at `https://www.dyndns.com/`.

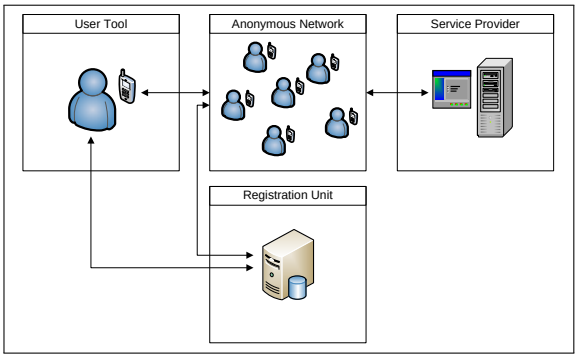



Figure 3.13: Intermediate anonymous network architecture

The application of session keys requires SPs to make public keys available. For high user trust it is best to publish them on a well known and accepted public key server, like the *MIT PGP Public Key Server*[34]. If the user tool wants to agree on a session key for a particular transaction, it generates a random string, requests the SP's public key from the key server and encrypts the string accordingly. In the next step the encrypted key is sent to the SP. To avoid direct contact, the anonymous network is used in the same manner as if pseudonyms are sent. The SP then decrypts the key with the corresponding private key. In this way, both parties are aware of the same key even though the SP does not know it's communication partner.

Malicious interstations could exchange the encrypted data packet and try to send SPs their own session keys. However, splitting session keys into different data packets leads to the result. that SPs either rerequest malicious packets (if they can explicitly determine affected ones) or reject entire communications. And as a reason that it is almost impossible for attackers to exchange every single data packet, successful agreements on a faked session key are most unlikely. Anyway, even if session keys would have been replaced, it is almost impossible for malicious interstations to decrypt user pseudonyms, because firstly they are split and secondly they are encrypted with an unknown session key. Thus, there is a very low probability that users could be harmed. For the sake of completeness there are a few mechanisms available that help to avoid those attacks. However, in order to keep the amount of computation on mobile phones as low as possible, they are not applied in this project.

At this stage, every component of the proposed protocol for protected and anonymous data transmissions and the underlying network are designed. Therefore, Figure 3.14, as an updated version of Figure 3.13, visualizes the final network architecture. Thereby, the added external key server is responsible to hold public keys of SPs.

**Interim Conclusion**

As a brief review and conclusion, the corresponding process sequence needs to be visualized. However, because of limited space the corresponding figure is placed in Appendix A.2. The there shown UML diagram specifies how the user tool sends a data message anonymously and encrypted to a SP[35]. Moreover, the second box represents not only one system but rather all participating mobile

[34] For more information it is referred to `http://pgp.mit.edu/`.

[35] How to technically transform XML pseudonyms into appropriate request messages on SP sides is not part of this project.

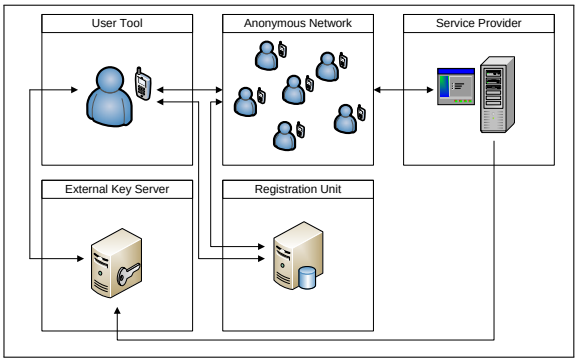



Figure 3.14: Final anonymous network architecture

phones, so that processes are carried out by different interstations. In case data packets are missing or corrupt, Figure A.2 also determines how users and SPs are able to rerequest them.

To complete this protocol design a few comments are necessary. Firstly, it needs to be mentioned that the mechanism used to interpret and forward data packets comes with the user tool itself. Thus, only mobile phones with corresponding installations are able to participate in the anonymous network, whereby taking part is voluntary. Users who do not want to use this network or to register at the registration unit are nonetheless able to securely send and receive encrypted packets. However, in this point to point communication no anonymity can be guaranteed.

Secondly, while the addressed mechanism automatically comes with the user tool, SPs by default are not able to interpret encrypted and split data packets. This is the reason why they need to integrate the proposed protocol that places itself between the communication and the online form. Whenever encrypted and split data messages arrive at the SP side, this protocol recognizes that it needs to encrypt and merge the packets so that the information can be delivered as a request message to the intended address (as if the data would have been directly sent through the online form).

The third comment relates to the trust factor. In order to enable high anonymity, dependency on two central stations can not be avoided. So, without using a central key server no appropriate encryption is possible. However, this project assumes the application of a well known and widely accepted public key server so that maximum support is provided. Besides this, the registration unit is required in order to keep anonymity on a high level; but if users do not trust this particular system they can easily bypass it, as stated previously.

The fourth and last comment refers to [56]. This source notes that there is an important aspect to keep in mind when applying services like DynDNS in UMTS networks. Thus, it needs to be guaranteed that the mobile providers' routers and proxies allow connection attempts from the internet; otherwise interstations can not be addressed by their DNS names and the solution is not applicable.

Finally, Subsection 3.2.3 specified requirements related to the design of the actual protocol, as reviewed in the beginning of this discussion. Table 3.6 shows to what extent they are fulfilled. It can be summarized that the protocol for protecting anonymous data transfers has been successfully

developed. Like addressed, the only external regulation so far is that SPs are required to integrate the proposed protocol because no standardized solution was applicable. Moreover, in Table 3.6 only the *lowest level of trust* requirement is marked as *mostly solved*. This is the reason, because the design of an anonymous network without any trust relations is not possible.

Table 3.6: Fulfillment of protocol requirements for protected and anonymous transmissions

| **Requirement** | **Status** |
|---|---|
| Encryption of both communication directions | *Solved*<br>Both parties apply one and the same session key. |
| Continuous and entire user anonymity | *Solved*<br>Hop counts prevent identification of sending station. No public key on user side required. Session keys transmitted over the anonymous network. |
| Encryption and key exchanges based on open standards | *Solved*<br>Application of session key cryptography and public key server. |
| Different encryption keys in every transaction | *Solved*<br>Change of session keys in every transaction. |
| Encryption as fast as possible | *Solved*<br>Asymmetric encryption only applied once (agreement on session key), all subsequent encryptions carried out with fast symmetric cryptography. |
| Availability of various anonymity levels | *Solved*<br>Possibility to choose different amounts of packets and thus required interstations. |
| Processing power as low as possible | *Solved*<br>Symmetric encryption works with low computational power. |
| Lowest trust relations in other parties | *Mostly Solved*<br>Trust in well known public key server and (if necessary) registration unit required. |
| Independence from underlying communication infrastructure | *Solved*<br>TCP/IP enables use of *DynDNS* as addressing scheme for e.g. UMTS and WLAN. |

### 3.3.4 User Tool

This subsection describes the design of the user tool. It starts with discussing other works that are taken into account during development to place the framework into a larger context. After this, the functional contribution of the proposed tool is identified and the results of the requirements analysis are taken into consideration. By maintaining the introduced distinction between user interfaces and user tool mechanisms all related functionalities are designed.

Subsection 3.3.4 concludes the entire design process of this project. It also finalizes the technical system architecture and the related process sequences. Together with various suggestions for different interfaces all provided information eases handover of the framework to programmers, who should be able to build the first framework prototype[36].

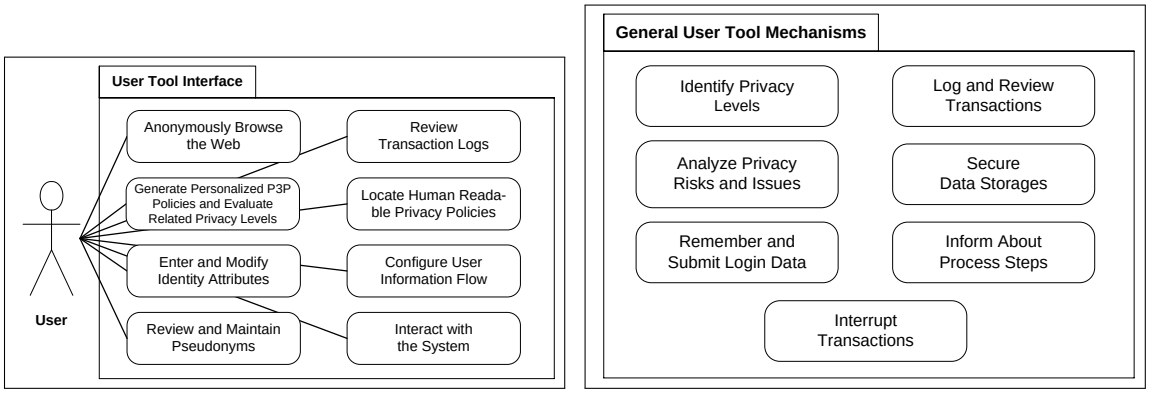
[36]Prototyping was one of the project objectives, but due to lack of time it proved impossible to carry out. However, it was defined as an additional objective that is not essential for this thesis to be a success.

### Reference Systems and Functional Contribution

Reviewing Section 1.3 and the beginning of Section 3.3, three systems were stated as references for the development process. However, with regard to user interface designs [12] does not provide any input. On the contrary, in [16] the authors designed a user interface to modify policy files. Even though the framework requires a similar interface, an entire adaptation is unfeasible. This is mainly because the solution was designed for large computer monitors rather than small mobile phone screens. Furthermore, the authors defined completely differing policies. A similar situation applies to [6] that also specified an interface. Primarily developed for mobile phones it is useful to get the design process started but the different overall approach again prevents adaptation. However, while both system are less helpful at this stage, they are of much more interest to be applied as discussion criteria in Chapter 5.

Figure 3.15 shows the contribution of the user tool by presenting functions that are derived from the requirements analysis of Section 3.2. Despite tight overlappings it has been attempted to divide those into two treatments of focus. Figure 3.15a lists all interactive activities that users carry out in collaboration with the user tool. All other processes that are not directly connected to interaction are then summarized as general user tool mechanisms in Figure 3.15b.

Because of redundancy reasons, mechanisms that belong to the previously designed protocols are neither shown nor discussed here. However, they nevertheless stay in closely connection.




(a) User interaction with the user tool

(b) General user tool mechanisms

Figure 3.15: User tool functional contribution

### User Interface

In Figure 3.15a eight ways of interaction with the user tool are shown. Like introduced earlier they are defined as *UI1* to *UI8*. These interfaces support users in the entire identity data life cycle that was addressed in Chapter 2. It is possible that based on their closely connection, two interactions are integrated and described together. Furthermore, the following discussion is wherever applicable complemented by screenshots of suitable interface designs; this provides a good basis for subsequent prototyping. In order to optimally display the interfaces on small mobile phone screens, large presentations make use of the so-called *landscape view* that rotates interfaces by 90 degrees.

### UI1: Anonymously Browse the Web

In order to achieve high anonymity not only transactions are carried anonymously. Moreover, web browsing is also carried out through the same network infrastructure. But this time data messages are neither divided into packets nor encrypted. However, the hop count data element is integrated to avoid privacy violations like tracking and to ensure compatibility with the designed protocol that guarantees protection of anonymous data transfers. These slight modifications to the data packet design of Figure 3.12 are presented in Figure 3.16.

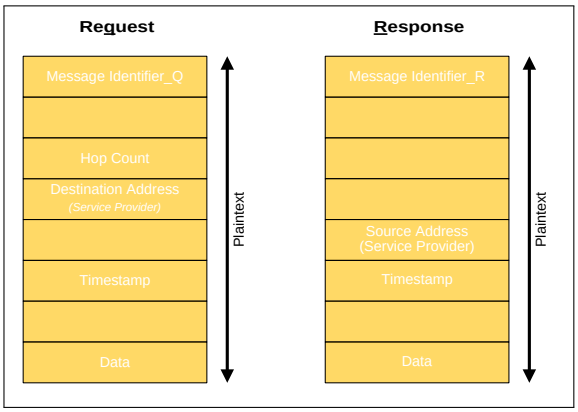



Figure 3.16: Data packet design for browsing activities

For this activity a particular interface enables users to browse the web in the usual manner. But in contrast to other browsers this one analyzes content of HTTP response messages before it displays any information on the screen (see procedural method in Subsection 3.3.2).

The development of such a browser in Java programming language is not challenging at all; all required classes and methods are available today. However, this project does not provide any code but shows one possible interface design in Figure 3.17.


Figure 3.17: User interface to anonymously browse the web

**UI2: Generate Personalized P3P User Policies and Evaluate Related Privacy Levels**
The interface of Figure 3.18 allows to generate personalized P3P user policies[37]. It drives users through a simple dialog that asks for all relevant specifications. Additionally, mouseover effects provide more detailed information about every single input. In the end the dialog allows to evaluate the related privacy level. The same interface also allows to review, modify and delete existing policies.

To support both, users with no P3P and XML knowledge and advanced ones,policies can be created in two different ways. The first is to go through the addressed interface dialog. The second is to directly use P3P XML syntax. These options are visualized by two tabs in the figure's upper right corner. After completion, the user tool automatically generates corresponding P3P policies and securely stores them on the mobile phones.

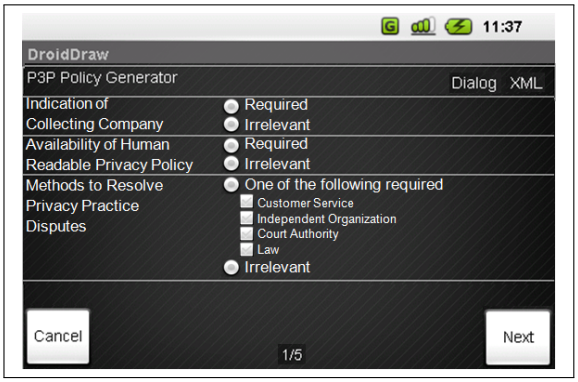



Figure 3.18: User interface to generate P3P policies and evaluate related privacy levels

With the goal of standardization this project assumes default privacy levels. Thereby, evaluation values range from zero to ten, as visualized in Figure 3.19. Zero indicates no privacy at all, an increasing level specifies higher privacy and the highest level is finally reached by ten. How to technically integrate those levels into policies is described during the design process of UTM1.

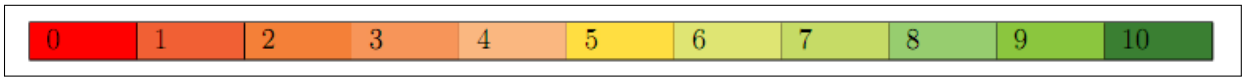



Figure 3.19: Standardized privacy levels between 0 (no privacy) and 10 (high privacy)

**UI3: Enter and Modify Identity Attributes in Local Databases and at Service Provider sides**
To enter and modify identity attributes the user tool provides an interface that is similar to the previous dialog. A table lists all possible elements on the screen, whereby they are categorized according to the P3P vocabulary. All data that is provided in this interface is stored in the databases according to the structure that was proposed in Subsection 3.2.3. Figure 3.20 shows an extract of the entire form.

In case users modify existent identity attributes a small checkbox on the right site is marked to indicate changes. When the *save* button is clicked the tool identifies all marked boxes. It then

[37]For reasons of space only the first page is shown here; the entire dialog is placed in Appendix A.3.5.

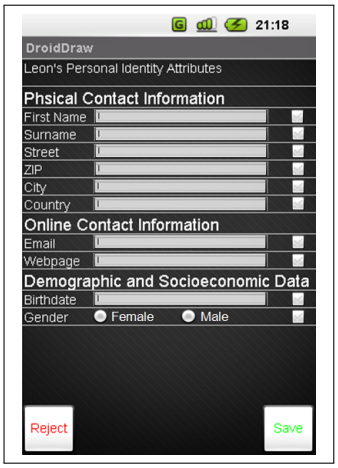



Figure 3.20: User interface to enter and modify identity attributes

reviews logs and stored pseudonyms to detect affected SPs. In the next step it considers related policies that are also stored in the logs in order to analyze specified ways of access. However, SPs are not required to define the exact *method of access*, the only requirement is to state *what* data is accessible. Thus the goal of the user tool is to identify whether the modified attribute is accessible or not. If so, it links the appropriate human readable policy to the summary shown in Figure 3.21. This helps users to quickly locate SP statements that describe possibilities for access and modification.

Figure 3.21 indicates that YouBuy and DressYourClothes allow modification of the affected attribute. For this, the *Manually* data fields are linked to the human readable policies. This interface also informs that Amazonas does not provide appropriate access for the modified identity attribute.


Figure 3.21: User interface that summarizes ways for identity attribute updates

**UI4: Review and Maintain Pseudonyms of Previous Transactions**
The next interface enables reviews and maintenance of attributes that have been disclosed to SPs by making use of the pseudonym files that were introduced in Subsection 3.3.3. When this interface is started the user tool automatically locates all available pseudonyms on the mobile phone. It then extracts relevant information from their filenames and displays a table as shown in Figure 3.22. For more detailed information a click on each line opens corresponding pseudonyms so that the disclosed attributes and the treated context become visible.

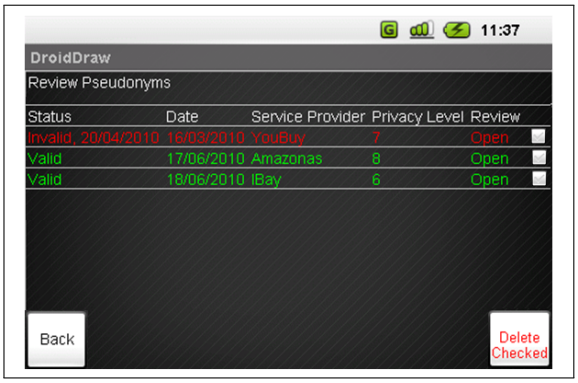



Figure 3.22: User interface to review and maintain pseudonyms

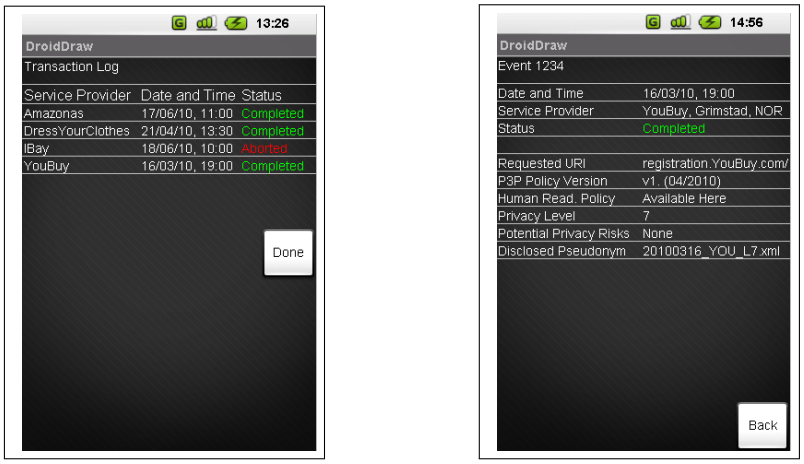
As addressed in Subsection 4.3.2, one and the same pseudonym is applied to subsequent and similar transactions for efficiency reasons. However, users may not want such reuse and identity updates in the databases could also make pseudonyms invalid. Thus, users are able to delete stored pseudonyms and attribute modifications automatically mark affected pseudonyms as invalid. This results in the situation that particular pseudonyms are not disclosed anymore.

**UI5: Review Transaction Logs**
The interface for reviewing transaction logs is intentionally oriented towards the structure of the *Microsoft Event Viewer*, to help many users with quick familiarization. It consists of two views whereby the displayed information is extracted from the logs that are stored on the mobile phones.

The first view presents a table that summarizes all transactions and indicates whether they were completed or aborted, as presented in Figure 3.23a. The second one is displayed when users want to retrieve more detailed information about a particular transaction, as shown in Figure 3.23b. Last but not least, a click on the *Disclosed Pseudonym* element opens the interface that is known from the previous discussion about pseudonym review and maintenance.

**UI6: Locate Human Readable Privacy Policies at Service Provider Sides**
The next interface enables localization of human readable privacy policies. It automatically extracts the required information from P3P policies of current transactions. According to the process of Figure 3.9, step one requires lots of information to be displayed and agreed upon by users. Thus, this interface is very useful for integration of an additional data element, as visualized in Figure 3.24. It allows users to directly browse addressed policies by simply clicking the particular notification[38]. This particular interface is displayed in every transaction and requires active user consent.

[38]The interface also indicates privacy levels and risks, but they lie out of the current discussion.




(a) Summary of all transactions

(b) Detailed view of a particular transaction

Figure 3.23: User interface to review transaction logs

**UI7 & UI8: Configure User Information Flow and Interact with the System**
With the goal of designing a transparent user tool and fulfilling the privacy design goals and requirements of Table 3.1, the user tool integrates a variety of notification messages. Moreover, transactions are completely based on user interaction and active consents. However, experienced users could feel bothered by information and confirmation messages. Therefore, wherever possible four different user choices are provided. Since there is no point to display any screenshot here, the following list briefly defines the possibilities.

- **Accept Once** - Agreement is only given for a particular transaction.
- **Always Accept for this SP** - Agreement is always given for a particular SP.
- **Always Accept this Combination** - Agreement is always given for a particular SP in combination with currently disclosed identity attributes.
- **Cancel** - Agreement is not given, the transaction is aborted.

In order to change past decisions and to personalize the frequency of information messages the user tool comes with appropriate functionalities that allow adjustments of every notification. However, there is again no need to visualize this functionality in a screenshot.

### Functional Mechanisms

Following the seven general user tool mechanism of Figure 3.15b are designed in the same way the interfaces have been discussed. As introduced, they are indicated as *UTM1* to *UTM7*. Together with the interfaces and the privacy and security protocols they comprise the entire user tool of the framework solution.

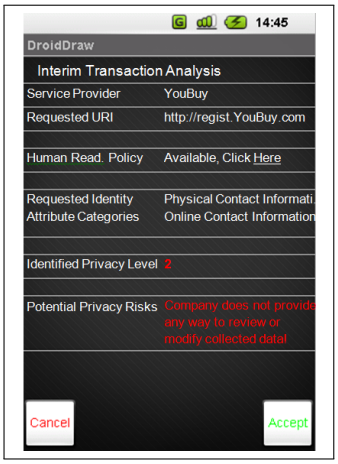



Figure 3.24: User interface to locate human readable privacy policies

**UTM1: Identify Privacy Levels**

While UI2 describes the interface to specify privacy levels, the discussions here adds two supplements. First, it clarifies how to technically integrate privacy level evaluations into the policies. Then it describes the mechanism to identify levels of upcoming transactions. A concluding comparison example makes UTM1 transparent.

To integrate level evaluations that are provided during policy generation, a new XML tag named $<PRIVACY\_LEVEL>$ is defined. It requires no modification of any XML namespace because the user tool is the only unit that needs to able to interpret this tag.

Granted that Carol creates a policy and evaluates it with a privacy level of 7. Referring to Figure 3.19, Carol thus assumes the SP to handle her identity data in a highly privacy-sensitive way. Carol defined that five requirements have to be met in order to reach the intended level[39].

- A human readable privacy policy is available.
- The collecting company is completely specified.
- Access to collected identity attributes representing contact information is provided.
- Collected attributes are only used for completion and support of the current transaction.
- The only identity attributes to be collected are *Name* and *address* that belong to the *user* data[40].

Transferring these specifications into the XML P3P syntax, the following policy is created. Listing 3.7 shows that the privacy level tag is added to the end; this way the P3P format is sustained.

[39] Specifying P3P policies and related privacy levels may be challenging for unexperienced users. Therefore, Appendix A.4.1 provides several sample evaluations that can be used as reference.

[40] The exact specification of identity attributes is just used for clarification purposes. By default, the proposed user interface only allows to specify data categories rather than unique identity attributes. The general idea behind this is to simplify policy creation and privacy level evaluation.

```
1 <POLICY name="Carols_P3P_Policy_Level_7"
2       discuri="*" <!-- human readable privacy policy available -->
3        xml:lang="en">
4 <ENTITY>
5   <DATA-GROUP>
6    <DATA ref="#business.name"/>
7    <DATA ref="#business.contact-info.postal.*"/>
8    <DATA ref="#business.contact-info.telecom.telephone.*"/>
9   </DATA-GROUP>
10 </ENTITY>
11 <ACCESS><ident-contact/></ACCESS>
12 <STATEMENT>
13   <PURPOSE> <current/> </PURPOSE>
14   <DATA-GROUP>
15     <DATA ref="#user.name.*"/>
16     <DATA ref="#user.home-info.postal.*"/>
17   </DATA-GROUP>
18 </STATEMENT>
19 </POLICY>
20 <PRIVACY_LEVEL> 7 </PRIVACY_LEVEL>
```

Listing 3.7: Carol's P3P policy that represents a privacy level of 7

The overall approach for identification of privacy levels is to compare current SP policies with all user policies on the mobile phones. The fact that those are written in a common syntax simplifies this mechanism. So, in the first step the user tool searches for the XML opening tag <POLICY> that indicates the starting point for the analysis. It then compares each specification line by line. If the user requirements are met by the SP specifications, it is registered in a comparison file. The same happens for mismatches. Each match or mismatch then increases a counter in that particular file. In the end the user tool is able to calculate the percentage match factor.

This process is carried out for all user policies so that finally the one with the highest match factor indicates the most likely privacy level for upcoming transactions. In case two files have the same percentage factor, the lower level is automatically chosen[41]. The identified privacy level is then displayed on the mobile phone as visualized in the interface of Figure 3.24. However, in case all match factors are below a particular limit that is individually adjustable by the users, no decision is made. Here, the user tool rather displays appropriate information so that users can decide whether to still carry out the transaction or not.

Applying this approach to Carol's (Listing 3.7) and YouBuy's (Listing A.1) policies, different comparisons are possible.

- Carol requires a human readable policy to be existent (line two). The tool locates the specification *discuri="http://www.YouBuy.com/P3P/disc.html"* and identifies that Carol's requirement is met.
- Carol requires the company's name to be defined (line six). The tool finds the same specification and identifies a match.

[41] The tool needs to be aware that not only 100 percent matches indicate agreements. In particular, some specifications reflect a hierarchical structure. Granted that a user defines that access to identity data belonging to physical contact information is required. The SP, however, specified that all collected identity attributes are accessibility. This obviously does not result in an one-to-one match but the hierarchy indicates that the user statement is still fulfilled. Those dependencies need to be clarified and integrated when coding the framework prototype.

- Carol requires the company address to be defined (line seven). The star indicates that a match is in place if any postal contact information is specified.

While each of these matches or mismatches increases the counter by one, the <DATA GROUP> within the <STATEMENT> tag is treated differently. Here, not only identity attributes are analyzed and evaluated. Rather it is required to also see the entire <DATA GROUP> tag as one overall requirement. This definition is assumed because identity data plays the main role in the entire framework solution. Thus, besides the normal counting a match or mismatch of the overall requirement counts two additional values[42].

Table A.3 in Appendix A.4 clarifies this complicated but effective evaluation. It also exemplifies the comparison of the two policies as briefly started here. The aim is to clarify UTM1 and to show the percentage match factor. Even though the identification of UTM1 is not 100 % precise, it aims to provide users with a quick indication and hint of the privacy level they most likely will be confronted with in the intended service request.

**UTM2: Analyze Privacy Risks and Issues**
For the analysis of privacy risks and issues current P3P policy specifications are joined with information stored in log files[43]. Based on predefined identification rules particular occurrences lead to alerts on the mobile phones.

**Analysis of Privacy Risks and Issues** Leon previously ordered items at YouBuy. The log file on his mobile phone states that he provided postal address and telephone number. In the current transaction YouBuy now requests Leon's identity attributes *birth date* and *identification number*. This alerts the user tool of possible privacy risks related to user data concatenation. It therefore displays an appropriate notification message on the screen. If Leon wants to progress, his active consent is required. In this way it is guaranteed that he is aware of the potential risk that could decrease the identified privacy level.

As address, to integrate this mechanism into the user tool, identification rules are necessary; the tool is required to *learn* privacy risks in advance in order to act appropriately. Thus, in addition to Appendix A.4.1 that provides typical privacy level specifications, Appendix A.4.3 shows several sample identification rules that specify occurrences of different privacy risks and issues. Those are the ones to be integrated into the user tool to enable adequate analyses.

As a conclusion, the identification rules are also the reason why these few paragraphs already complete the design process of UTM2. The entire functionality is simply based on particular rules that allow comparison of current transactions with previous ones. In case an identification rule matches, corresponding notification is added to the interface of Figure 3.24. As seen, active user consent is then required to continue transactions.

[42] Briefly reviewing Carol's specifications: User data *name* and *address* are the only attributes she allows to be collect. If a SP requests *name*, *address* and *telephone number* a mismatch of the overall data group is in place. In case only *name* and *address* are collected a match exists. In the last event, a SP collects only *name*; this also matches with Carol's specifications.

[43] Log files were not discussed in detail so far. They simply log all transactions into data files that are securely stored on the mobile phones. Besides other data, they log names and addresses of SPs that requested data. They also contain information about what particular attributes users disclosed to different SPs.

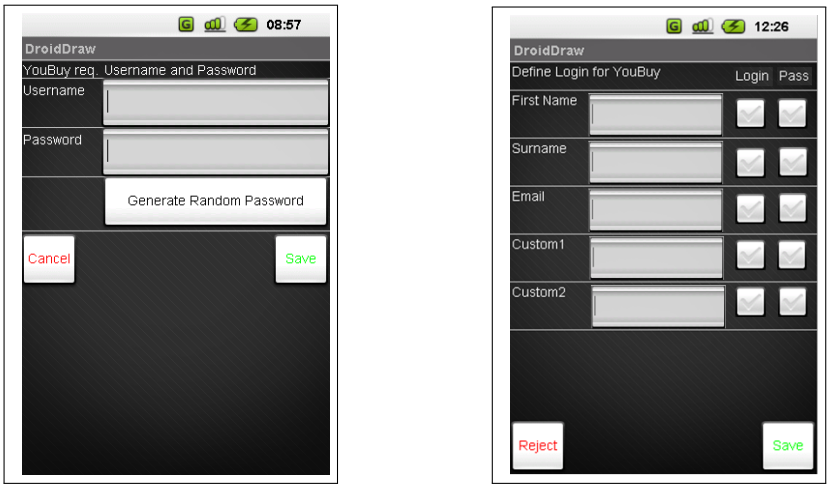
**UTM3: Remember and Submit Login Data**

Once again, a major advantage of IDMS infrastructures is the support for remembering large amounts of account data, by means of logins. In the framework solution UTM3 is responsible for this task. It provides an interface that automatically opens when usernames and passwords are requested (see Figure 3.25a). This is necessary because those particular attributes are SP dependent and not defined during initial configurations.

The same interface also offers a simple way to generate secure passwords that fulfill the most important complexity rules. Closing the interface leads to automated progressing and saving of the attributes in the local databases, whereby descriptions comply with the P3P vocabulary. The data is also internally linked to the appropriate SP to achieve unique mappings. In cases where users request services from SPs they are already registered a user account, they may be asked for login data. The user tool handles those requests just the same way as it does for other identity attributes - it automatically selects them form the database.

However, not all SPs specifically ask for usernames and passwords. In fact, some of them use for example email addresses as usernames. Others do not allow users to specify passwords and rather send out generated ones after registrations. Thus, it could be the case that the tool fails to locate login data in the databases. To react properly, the user tool comes with a further mechanism.

A second interface therefore enables the review of pseudonyms and the definition of login data. It lists pseudonyms as previously presented in Figure 3.22 to show all disclosed attributes that are worth to be considered as logins. By selecting checkboxes (see Figure 3.25b) users can specify usernames and passwords. A copy is then stored under appropriate paths in the databases. Finally, the interface also provides two additional data fields (*Custom1* and *Custom2*) that allow the definition of individual items that are not included in any disclosed attribute.




(a) Interface to submit username and password

(b) Interface to define login data

Figure 3.25: User interfaces to remember and submit login data

Many SPs integrate registration functionalities into their login pages. Thus, it is highly possible that users click on buttons that automatically redirect them to those fields. However, they probably do neither have any login data because they did not register so far nor there is anything useful stored in the databases. This is the reason for the the *Cancel* button in the interface of Figure 3.25a. If users click on it, the login page with the input forms is displayed on the screen. For privacy reasons an exploratory information message is then shown, that notifies that every identity attribute currently provided is not be treated privacy-aware by the user tool. However, this possibility is mainly provided in order to allow user to choose appropriate registration buttons; and as soon as new website content is queried that requests identity attributes, the user tool takes over responsibility again.

**UTM4: Log and Review Transactions**
Discussing log files there are three aspects to take into account.

1. Information to be logged.
2. Ways to store log files.
3. Possibilities to review logs (here, with focus on automated rather than user-oriented reviews).

Log files, as address a couple of times before, help to make the entire system transparent. Therefore, transaction and system related data is logged as granularly as possible. With reference to Figure 3.23 the following items need to be included.

**System Information**

- User access to identity attributes in local database.
- Modification of identity attributes in local database.
- Generation and modification of policies on the mobile phone.
- Modification of pseudonym validity on the mobile phone.
- Deletion of pseudonyms from the mobile phone.
- Modification of the system configuration on the mobile phone.

**Transactional Information**

- Data and time of disclosure.
- Contacted SP.
- Transaction status (completed or aborted).
- Requested URI.
- Applied SP policy.
- Related Human readable policy.
- Identified privacy level.
- Analyzed privacy risks and issues.
- Disclosed pseudonym.

Regarding storage of log files there are two different approaches practicable. Either logs are written into separate data files or, the more coherent solution, they are integrated into the local databases. However, this decision is dependent on the programmer and the applied databases.

Last but not least, the review of logs is supported by the interface UI5 (see Figure 3.23). While this is designed for users, it is just as important that the user tool itself is also able to read and interpret logs. In particular ,it is required to support analyses of privacy risks and issues in UTM2. Thus, in order to provide input for the identification rules presented in Appendix A.4.3 the user tool is required to answer the following questions, mainly with the help of the P3P vocabulary.

- When did the user disclose what attributes to which SP?
- Which SP policy was applied? How where recipients, retention, access, disputes and purpose specified?
- What identity attributes were disclosed when requesting which particular URI?

**UTM5: Secure Data Storages**
Databases on the mobile phones store the critical information identity attributes, pseudonyms and transaction logs. This data in the wrong hands can result in serious consequences, as addressed in Subsection 2.2.2. Sufficient care and security is thus unavoidable.

Searching for an appropriate protection, this project makes two assumptions. First, it requires file based encryption to be part of the mobile phone operating system, here Android[44]). Second, for the same reasons like stated during data packet encryption, it assumes that symmetric cryptography is used. Based on these two facts the main focus lies in securing symmetric encryption keys.

Simply password protected placing them on mobile phones would be one possible solution. However, the achieved security in this case depends to a great extent on the password quality. A better way is related to the general key server approach that was described in Subsection 3.3.3. The concept with this server that holds encryption keys of all participants is that every time users require access to encrypted data on their mobile phones they need to request the necessary key.

In order to avoid misuse and to increase security keys on the server are not stored in the default key server way but rather symmetrically encrypted. Thereby, the corresponding Key Encryption Key (KEK)s are saved on the mobile phones - each phone owns an individual KEK. This guarantees that only allowed devices are able to decrypt keys that they received from the key server.

Up to now the benefits of a key server did not become clear to the widest extent, because symmetric data encryption keys directly stored on the mobile phones could lead to almost the same result. However, when talking about device loss the situation changes. This is to say, in order to appropriately secure data after a phone is lost, encryption keys for the critical data are only kept alive during program runtime and automatically destroyed when the user interface is closed. This means, that every access to encrypted information requires communication with the key server. With this approach users are allowed to immediately block access to the affected key, or in more general to the server. Then, even though finders will have the KEK stored on the phone, they are not able to request the necessary key from the server and thus subsequently prevented from data decryption.

[44]Today, encryption functionality is not included in the system configuration. However, there are already corresponding applications available, like e.g. *MyStash file encryption* (see `http://www.beysoft.com/mystash.aspx`. This shows that there is a demand for encryption that results in a high probability that appropriate functionalities will be included soon.

While there are different standardized key server solutions on the market, none of them is adoptable for this concept because of it's particular requirement to store encrypted rather than plaintext keys. However, an individually configured server leads to an effective and efficient solution based on the two security requirements *mobile phone with KEK* and *access to key server*. Only if both requirements are met encryption key requests and data decryption can be successfully carried out. As a conclusion, sometimes device loss is not recognized immediately; here, thieves could probably have enough time to decrypt data. To address this issue, a third requirement (e.g. identification with passwords) may be helpful but would result in additional login data to remember. Thus, the related usability aspect is the reason why this approach is currently not suggested for the framework solution.

**UTM6 & UTM7: Inform About Process Steps and Enable Transaction Interrupts**
The last user tool mechanism briefly supplements user interaction aspects to UI7 and UI8. In particular there are features necessary that permanently inform users about the current system state and provide various notification messages, a summary before finally disclosing identity attributes and ways to interrupt transactions. However, no screenshots are provided here because the realization is strongly dependent on the prototype that will be designed after project closure.

The most helpful extension in this context is a progress bar. Being permanently visible it informs about current process steps, interacting SPs and identified privacy levels. Whenever necessary, additional data can be added, but it is important to realize that too much information could confuse users. However, generally it can be said that the more data is displayed the more transparent are system and transactions; at least for unexperienced users this leads to higher well-beings. But as mentioned in UI7 and UI8 the tool also provides several ways to configure the frequency of all notifications in order to personalize system behavior.

Besides this progress bar, the privacy requirements *user-centering*, *transparency* and *user consent* (Table 3.1) require to enable users with ways to immediately abort processes and entire transactions. Therefore, every critical user interface in the previous design process was equipped with buttons that allow continuation and cancellation. In addition, UTM6 and UTM7 integrate an overall emergency exit button that is easy to locate, self-explanatory and permanently visible. This particular button then provides users with a way to interrupt all running tasks, processes and transactions, if needed.

#### Interim Conclusion

This subsection equipped the procedural method with the privacy and security protocols and integrated it into a user tool. Thereby, general ideas and approaches of reference works were adapted wherever applicable. Furthermore, it was necessary to add a subsystem to the technical system architecture that is responsible for data encryption on the mobile phones. The resulting overall system architecture for this is presented in the next section.

As a conclusion, during the design process main focus was put on fulfilling the privacy requirements of Table 3.1, the functional requirements FR5 and FR6 and the user tool interface and mechanism related requirements that were reviewed in the beginning of Subsection 3.3.4. At this stage it can be stated that all of them are successfully met.

### 3.3.5 Conclusion

Section 3.3 presented the framework design process. It started with finalizing the procedural method, then designed the required privacy and security tools and finally integrated all aspects into the framework user tool. Thereby, the privacy requirements and design guidelines of Table 3.1 were constantly consulted. Referring Subsection 3.2.5, all six functional requirements are successfully met, as shown in the following section and the Chapters 4 and 5.

## 3.4 Framework Summary

With the help of two PM design guidelines (see Table 3.1) the framework is now complete. It's parts *procedural method*, *privacy and security protocols* and *user tool* are defined and all necessary internal and external components specified. Moreover, research questions two and three were answered. In contrast to almost all IDMS solutions existing today this framework does not apply preconfigured, fixed pseudonyms but rather follows an approach that is not available for the mobile area so far. That is to say, it combines individual identity attributes to pseudonyms for particular transactions. The unique benefit of this is that always only those attributes are disclosed that are actually required.

The current Section 3.4 briefly summarizes the framework development. It addresses the technical system architecture, reviews user scenarios and functional requirements and gives an outlook into infrastructure and technology and to SPs. Thus it works as an introduction for Chapter 4 in that the technical facts of the three framework parts are reviewed. That chapter also clarifies subsequent steps that are required to finally deploy the proposed framework. This is also the reason why Section 3.4 is intentionally kept short.

### 3.4.1 Technical System Architecture

The final technical system architecture consists of four subsystems. As visualized in Figure 3.26 those are the identity agent, the identity proxy, the (anonymous) network and the SP. As clarified during development, identity agent and proxy together represent the user tool.

Every transaction is initiated by the identity agent in form of the user interface. The identity proxy splits messages, encrypts data packets and sends them into the anonymous network over UMTS or WLAN channels. Thereby, the anonymous network consists of four main components. Various interstations are responsible to forward and deliver data, the registration unit administers and provides addresses, the external key server holds SP public keys and the internal key server manages and provides encryption keys for protected data on mobile phones. After going through the anonymous network, data packets finally reach the forth and last subsystem, the SP. For responses those apply the same process sequence as was used for the request, but in reversed order.

### 3.4.2 User Scenario and Functional Requirement Review

In the beginning of Chapter 3 two user scenarios were defined. Those were constantly reviewed during the framework development in order to emphasize on user benefits and to show hands-on problems. Moreover, related user scenario questions in Subsection 3.1 presented typical privacy and security problems. At this stage it can be stated that the framework solution implements all

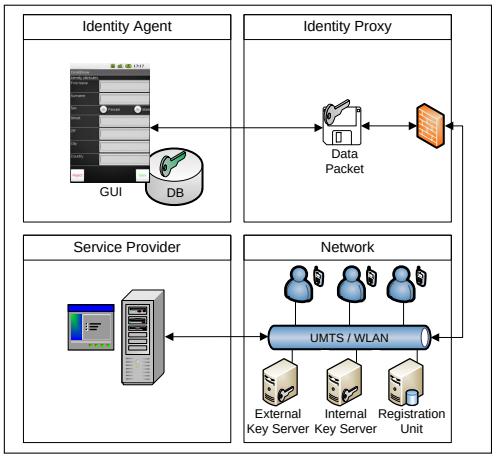



Figure 3.26: Finalized technical system architecture

derived specifications in a sufficient manner. If Carol and Leon follow the presented advices and make use of the designed systems, simplified and privacy-sensitive handling of identity data can be guaranteed.

In addition to the user scenarios, Chapter 3 worked out further specifications that together with the comparison of centralized and decentralized architectures contributed to six functional requirements. As a conclusion all those functional requirements are successfully met. The related technical implementation is described in Chapter 4.

### 3.4.3 Infrastructure, Technology and Service Providers

In Subsection 3.2.1 the framework's underlying infrastructure and applied technology were analyzed and requirements for SPs worked out. This subsection here briefly reviews those results and refines and completes the stated requirements and specifications. The discussion is thereby based on information that was derived during the framework design process.

#### Infrastructure and Technology

The proposed system architecture requires the integration of an *external* key server that is responsible for managing and maintaining SP public keys. It needs to be accessible by the user tool and SPs. Like addressed earlier, one of the most practical solutions for this system unit is the *MIT PGP Public Key Server*. The main reason for this assumption is that the system is well known and has been extensively tested and verified. This helps to successfully establish high levels of user trust.

In addition to that system unit an *internal* key server is required to hold protected encryption keys for data on mobile phones. In contrast to the external server accessibility for the internal one is limited to the user tool. Unfortunately, the project's particular specifications have shown that there is no appropriate system unit available so far. This makes it necessary to install an individual solution based on the requirements and specifications that were defined during the framework design.

The third and last additional system component in the underlying infrastructure is the registration unit. The requirements for accessibility are the same as for the internal key server. There are two implementation ways feasible that differ in their aspects dealing with user trust. The first possibility is to integrate the registration unit directly into the internal key server; the second way is to build up a standalone system. From security aspects the later solution is recommended. However, it could be the case that users feel more comfortable if they are required to trust only one system for both functionalities. This is thus a fact that needs to verified in field tests during prototyping.

As a conclusion, both the internal key server and the registration unit are required to be available to all participants over the internet. However, as soon as systems are opened to the web, protection on high levels is crucial. Here, the most suitable solution is probably to comply with the specifications of the *Jericho Concept*[45] rather than sticking to the parameterized security approach. On the other hand, there is no need to individually secure the external key server because this is the owner's responsibility (assumed that the MIT PGP Public Key Server is used). However, all those security aspects are just brief remarks, related discussion is not part of this project.

### Service Providers

The users' communication partner in this framework is always a SP. During design it was tried to make as few restrictions to them as possible. The reason for this is that every additional requirement can lead to the result that the solution is refused. However, it was not possible to avoid two particular specifications. In addition to them, the next chapter clearly states which preparation steps are required to be carried out on SP sides.

The first specification is thus that the anonymous network requires data messages to be split, encrypted, decrypted and joined on both communication sides. While this mechanism is directly integrated into the user tool, there is unfortunately no applicable standardized external protocol available so far. This makes it necessary for SPs to implement the proposed security protocol as a middleware that allows to carry out the required tasks.

Secondly, SPs need to upload their public keys to the external key server to allow agreements on session keys. However, this task should not be a challenge at all because SPs usually already own those keys for other or the same purposes. So, it only needs to be ensured that appropriate keys are made available on the *external* key server.

[45] *Jericho* is the term for a security project carried out by the Open Group. The goal is to dissolve unnecessary perimeters and rather secure each subsystem in itself. For more information it is referred to the project website at `https://www.opengroup.org/jericho/publications.htm`.

# CHAPTER 4

# Results

Chapter 4 briefly sums up the project results and provides a basis for the discussion in the next chapter. It describes what has been developed according to the framework parts procedural method, privacy and security protocols and user tool. Validation of them is then mainly carried out in the next chapter.

For now and based on the summary of Section 3.4 it can be stated that the framework design and development was carried out according to the requirements and specifications. Thereby, all functional and non-functional requirements were implemented. Furthermore, all twelve specified Privacy Design Requirements were successfully met.

To ease reading, the structure of Chapter 4 is aligned to the development strategy of Figure 1.1. It starts with a review of the background research, presents the results of the requirements and specification process, shows the design and development, and finally specifies the handover of the theoretical framework solution to programmers. This clarifies subsequent steps that are necessary in order to further develop and deploy the proposed system.

## 4.1 Background Research

This project started with analyzing the research areas within IM and PM in order to establish basic project knowledge and the theoretical background. It was learned that there is an increasing demand for privacy aware IDMS solutions - especially for the mobile sector no appropriate system has been widely adopted so far, although the need is strongly growing. But permanent connections to fixed environments in traditional IDMS infrastructures makes those unsuitable for roaming users. This is also the reason why today's most known IDMSs with their central storages fail to be integrated into the mobile area.

Even though a few mobile solutions have already been designed, they all come with remarkable disadvantages. Thus, they either just work with particular, preconfigured services or they provide only fixed pseudonyms from those the most suitable one is selected in every transaction[1]. Other solutions require modified SIM cards which makes large rollouts too expensive. Some systems are based on central registrars that host all identity attributes and thus require high user trust. And finally, a few approaches only work with static certificates that represent single pseudonyms for all transactions.

[1]This mechanism prevents to work in high privacy because it is most likely possible that unnecessary identity attributes are included in chosen pseudonyms.

Motivated by those research results the project aim was to design a framework for IM and PM on mobile devices that mitigates the identified disadvantages. In contrast to other approaches, dependencies on third party resources were required to be kept as low as possible. Moreover, decentralized storing of identity attributes was one key goal while designing the framework parts *procedural method*, *privacy and security protocols* and a *user tool*.

Analysis of different IDMS architectures led to the conclusion that most of them nowadays work with a combination of users, SPs and Identity Providers. However, the Identity Providers require trust relations. Thus, mainly based on the framework pillars *Security*, *User Trust*, *Cost Efficiency* and *Ease of Use* (Subsection 2.1.2) it is more suitable for this framework to hand over appropriate responsibilities to users.

With the goal to integrate privacy awareness into the framework, corresponding system solutions within this research area were analyzed. By identifying major privacy issues, typical PM Design Pitfalls and common Design Goals, particular *Privacy Design Requirements* for the framework were specified. Moreover, widely accepted PM Design Guidelines were chosen.

This background research allowed satisfaction of the project objectives (Subsection 1.2.2) and clarification of the contribution (Section 1.5). Based on all collected information it was possible to define framework requirements and specifications as shown in the next section.

## 4.2 Requirements and Specifications

In the second step requirements and specifications were defined. Based on the background research the goal was to identify the framework parts *procedural method*, *privacy and security protocols* and *user tool*. For the following design and development it was necessary to provide all specifications in a detailed manner.

### 4.2.1 Procedural Method

The procedural framework method describes the overall process sequence that is provided by the solution. It integrates IM and PM functionalities directly into everyday work flows in order to support users with handling and application of privacy-sensitive identity data in online transactions. As a reason that the solution was supposed to cause as little additional labor as possible, the procedural method aimed to be smoothly integrated into the process of web browsing.

The general idea is based on a modified browser that handles entire transactions (including browsing) completely anonymously. It's key feature is to automatically recognize identity attribute requests and to make related data forms invisible for users. Instead, it queries the SP data management practices and analyses whether affected users agree on it or not. In case agreements are established the browser initiates a well structured process sequence that takes users through an interactive dialog and allows simple and privacy-aware processing. In addition, a user tool as part of the framework enables administration of privacy-sensitive data that is stored on the mobile phones.

Concluding, the four key processes that are required for this method are briefly reviewed in Figure 4.1.

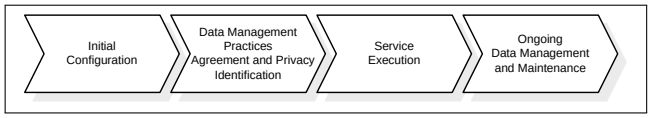



Figure 4.1: Review of the key processes related to the procedural framework method

### 4.2.2 Privacy and Security Protocols

The protocols are those framework parts that support the procedural method. Three related requirements were specified as reviewed here. It is important that even if they are separately defined, they still can be covered by one and the same protocol design.

**Request, Verification and Agreement on Data Management Practices** - SP policies that define data management practices need to be automatically requested and matched against user policies[2]. Comparison specifies whether agreements exist or not. For this automated process standardized language (vocabulary) between all participants is required.

**Unique Specification and Automated Selection of Requested Identity Attributes**[3] - Requested identity attributes need to be automatically selected from databases on the mobile phones. Here, a standardized language for requested and stored attributes is again useful. The user tool is required to adopt specifications from the policy vocabulary and to apply them when storing data in the internal databases.

**Protection of Anonymous Data Transfers** - Entire communications need to be secure and anonymous - SPs are not allowed to determine sending stations. Thus, a corresponding anonymous infrastructure that prevents interstations from reading data messages and identifying sources. Moreover, for flexibility reasons participation should be voluntary.

### 4.2.3 User Tool

The third and last framework part is a user tool. It represents the central component of the framework as being the interface to users. The tool was defined to integrate the procedural method and the privacy and security protocols into a user-centered system architecture. As an application on mobile phones that run with the Android platform it was supposed to provide functionalities to anonymously browse the web and to apply and handle identity data in privacy-sensitive ways. Moreover, the user tool was specified to consist of the two subsystems *identity agent* and *identity proxy* as shown following.

**Identity Agent** - Needs to provide various user interfaces that allow interaction with the system. The agent has to enable user-centered, privacy-sensitive handling of transactions and administration of personal identity attributes. It for example is required to support web browsing, user policy generation, identity attribute storage in the local databases and reviews of past transactions and processes. By permanently notifying users about critical process steps the agent has to guarantee transaction and user tool transparency.

[2]User policies represent individual specifications on how users allow SPs to handle privacy-sensitive data.
[3]This particular requirement is more likely a user tool mechanism. But based on the closely connection to the first requirement it was treated in the protocol context.

**Identity Proxy** - Is required to be responsible for processing of all internal mechanisms and establishment of communication over UMTS and WLAN channels. Different functionalities need to allow (amongst others) to securely store privacy-sensitive data on the mobile phones and to immediately interrupt transactions and tasks. The proxy has to identify privacy levels by matching evaluated[4] user policies against the ones from SPs. It is also required to join current policies with information about past transactions in order to analyze preconfigured privacy risks and issues[5].

## 4.3 Design and Development

In the third and last project step the framework was designed and developed according to the requirements and specifications of the previous task. It can be stated, that all functional and non-functional requirements were met and related functionalities integrated into the solution, as shown next.

### 4.3.1 Procedural Method

During development the requirements and specifications for the procedural method were implemented. Thereby, the entire system is designed to work based on data management policies. On SP sides those need to be publish for each website content that requests identity attributes. On the other side, users have to create one ore more policies that reflect specifications on how they allow SPs to handle identity data collection and processing. The following four figures review the procedural framework method in a simplified design. Discussion of some related aspects is then continued in the next two subsections.

Figure 4.2 starts with showing steps that are carried out before users are able to apply the main IM and PM functionalities for privacy-sensitive transactions. Therefore, users store identity attributes, generate user policies, evaluate related privacy levels and anonymously browse the web. The user tool then automatically recognizes identity attribute requests.

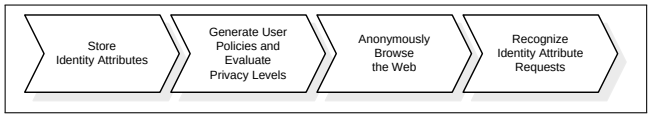



Figure 4.2: Initial system configuration and web browsing on user side

Next, the user tool blocks online forms[6], negotiates policy agreements and identifies potential privacy levels (Figure 4.3). It also analyses possible privacy risks and issues and subsequently selects requested identity attributes from the local databases. Those are then integrated into pseudonyms, split, encrypted and anonymously sent to SPs. In addition, the initiated transaction is logged for review purposed.

[4]Evaluation by means of individual privacy level specification (see Subsection 3.3.4).
[5]It was discussed that this functionality requires preconfigured identification rules.
[6]Online forms are blocked because identity attributes are rather directly sent to SPs than filled out by users.

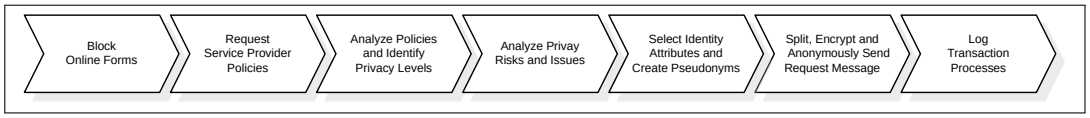



Figure 4.3: Service execution on user side

When SPs receive request messages they decrypt and merge the related data packets (Figure 4.4). They then interpret and process request messages. In the next step, they create appropriate response messages, split, encrypted and anonymously sent them back to the user.

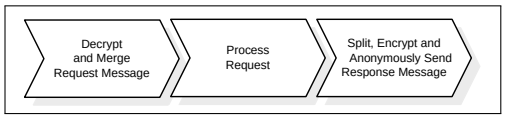



Figure 4.4: Request processing on Service Provider side

In the last step the user tool interprets the SPs response message and displays corresponding information on the mobile phone (Figure 4.5). Regardless of this, a separate interface provides ways to easily modify disclosed identity attributes in the local databases and on SP side.

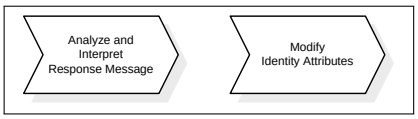



Figure 4.5: Response processing and ongoing data management on user side

### 4.3.2 Privacy and Security Protocols

Without repeating those aspects that were already reviewed in Subsection 4.2.2 the next three paragraphs sum up the key facts of the protocol development.

**Request, Verification and Agreement on Data Management Practices** - Agreements on data management practices are entirely based on the P3P technology and a project individual extension of it's vocabulary. Furthermore, an additional data element in user policies reflects personal privacy level evaluations. SPs need to modify HTTP headers of those websites that request identity attributes and publish P3P reference files and policies. The user tool then automatically requests this information that is necessary to agree on data management practices. The related comparison of user and SP policies is achieved by searching for particular XML tags and matching included data elements. In result, that user policy with the highest match factor indicates the most probable privacy level for the intended transaction. However, if all factors are below a personalized limit, appropriate information is displayed on the mobile phone that allows users to decide whether they still want to disclose identity attributes or if they prefer to cancel the intended transaction.

**Unique Specification and Automated Selection of Requested Identity Attributes** - This protocol is also based on the extended P3P vocabulary. It applies this to specify storing paths of identity attributes in the designed database structure on mobile phones. When SPs request identity data the user tool extracts relevant information from the applied policy and automatically selects stated attributes from the local databases. If items are missing a message is displayed that allows users to provide them in time. Otherwise, the selected attributes are stored in XML pseudonyms whose format is similar to the one of P3P policies[7]. Moreover, for efficiency reasons the user tool reviews logs in order to identify situations in that users already disclosed requested identity attributes to currently connected SPs; if so, past pseudonyms are reused. In the last step, identity attributes are handed over to another protocol that is responsible for the following transmission.

**Protection of Anonymous Data Transfers** - With the goal to securely and anonymously send data, an artificial network infrastructure was defined. It consists of all mobile phones with the framework user tool application. In the proposed anonymous addressing scheme participating interstations are defined by static DNS names that are managed by a central registration unit. Data messages that are going to be sent are split into packets with a predefined format. Thereby, the number of packets is adjustable to desired anonymity and security levels. Each packet is then encrypted with a session key and sent to another interstation. For anonymity reasons the packets integrate a particular data element that specifies whether interstations need to forward or directly deliver them. Regardless of this, these stations log particular transaction details that allows reuse of the communication paths in reverse order during responses. Recipients finally decrypt and merge the data packets in order to interpret related content[8].

### 4.3.3 User Tool

During user tool design and development the procedural method and the privacy and security protocols were integrated into an overall application. Various interfaces were designed that support the entire data life cycle and technically realize all previously described tasks. Additionally developed functional mechanisms are then responsible for the internal processing. Whenever possible, screenshots of system designs were presented to ease framework handover to programmers.

The following two lists briefly summarize the designed user tool interfaces of the identity agent and the functional mechanisms belonging to the identity proxy.

**User Tool Interfaces (Identity Agent)**

- **UI1** - Allows to anonymously browse the web through the proposed artificial network.
- **UI2** - Enables to generate user policies and evaluate related privacy levels.

[7] The idea behind this format is to reuse the P3P namespace.

[8] There are two important points to note. First, SPs so far are required to implement the designed security protocol that enables them carrying out related tasks. Second, the proposed network infrastructure works on a *peer-to-peer* approach. Thus, if users want to take part, they in return need to support anonymity of other participants.

- **UI3** - Supports entering and modification of identity attributes into the local databases. Provides a functionality that in cases of attribute updates initiates the user tool to review and display human readable policies in order to provide users simplified ways for modification on SP side.
- **UI4** - Allows to review and maintain pseudonyms, provides a functionality to block their reuses and indicates whether pseudonyms are invalid or not.
- **UI5** - Enables user-centered reviews of past transactions.
- **UI6** - Supports to easily and quickly locate human readable policies on SP side.
- **UI7 & UI8** - Allow to configure user information flow by adjusting the frequency of notification messages on the mobile phone.

**Functional Mechanisms (Identity Proxy)**

- **UTM1** - Identifies privacy levels by matching user demands against SP policies.
- **UTM2** - Analyzes privacy risks and issues by joining current SP policies with log file information and matching the results to preconfigured identification rules.
- **UTM3** - Helps in remembering and submitting login data, because usernames and passwords are SP dependent and thus not initially stored in the local databases. Provides a functionality to either specify those items on demand or select them from other already disclosed identity attributes.
- **UTM4** - Logs transactions and automatically reviews stored information in order to support UTM2.
- **UTM5** - Applies symmetric key cryptography to secure data storages. Encrypts corresponding keys with KEKs and publishes them on a key server. Stores KEKs directly on the mobile phones so that in cases of device loss communication to the server is blocked.
- **UTM6 & UTM7** - Integrate a progress bar to make system and transactions transparent. Add permanently visible emergency exit button that allows immediate interrupts of entire transactions.

## 4.4 Subsequent Steps: Coding, Testing and Deployment

According to Figure 1.1 this project concludes with the design and development of a theoretical framework solution. Section 4.4 now covers the three subsequent process steps *Coding*, *Testing* and *Deployment*. It shows all tasks that need to be fulfilled in order to finally deploy the proposed system. Thereby, it divides the activities into three areas of responsibility *Programmers and System Architects*, *SPs* and *Users*. This clarifies the handover of this project to programmers.

### 4.4.1 Programmers and System Architects

Coding, testing and deployment are carried out by programmers and system architects. They are required to realize the user tool with all designed mechanisms and interfaces, install additional system units into the infrastructure and roll out the entire system. Therefore, various figures, tables, UML diagrams, interface suggestions, process sequences and system architecture designs aim to support these processes and especially the handover of the theoretical framework as best as possible. Moreover, all proposed interface designs were created with the *DroidDraw* application[9]. The main benefit of this tool is a functionality to export designs and subsequently important them into the Eclipse programming environment. This way easy proof of concept emulations of the intended solution are feasible.

The following entities need to be treated by programmers and system architects according to the project specifications and the related requirements. In this context it is important that the extended P3P vocabulary does not require any modifications of the default XML namespace - it only needs to be applied to policy specifications, as stated in Subsection 3.3.1.

**Encryption Protocol** - The protocol that allows splitting and encryption (and vise versa) of data messages needs to be developed. As a middleware between SPs and published services it transforms XML pseudonyms into POST request messages and forwards them to specified URIs. Thereby, programming includes implementation and integration of the proposed data packet designs (Figures 3.12 and 3.16) and analysis of ways to realize the aforementioned data transformation. This protocol then needs to be distributed to SPs.

**User Tool** - The user tool needs to be programmed. This covers all proposed interfaces and mechanisms and the integration of a local database according to it's design structure (Figure 3.10).

**Internal Key Server** - The internal key server needs to be installed. Thereby, standard key server functionalities can be adopted and adjusted to the project specifications.

**Registration Unit** - The registration unit also needs to be installed according to the project specifications.

**Solution Testing** - Programmers and system architects are required to test the entire solution. Thereby, main criteria are the framework specifications of Table 3.1.

### 4.4.2 Service Providers

The framework solution requires participation of SPs that need to prepare the following items.

**Encryption Protocol** - The developed security protocol to split and encrypt (and vise versa) data messages needs to be integrated.

**Data Management Practices** - P3P reference files and policies need to be created and published.

[9] DroidDraw is an interface editor that helps to create designs for the Android platform. See `http://www.droiddraw.org/`.

**Human Readable Policies** - Human readable data management practice policies need to be published[10].

**HTTP Headers** - HTTP headers of those websites that request identity attributes need to be modified i order to specify the location of the P3P reference file.

**Public Keys** - Public keys for agreements on session keys need to be published on the external key server[11].

### 4.4.3 Users

There is not much preparation for users to do. The two main aspects to consider are the following.

**User Tool** - It needs to be ensured that the user tool application is installed on the used mobile phone.

**Device Registration** - Mobile phones that want to participate in the anonymous network need to be registered at the *DynDNS* service[12].

Those two aspects on user side also conclude the solution roll out. There is no need for more modification or adjustment of the underlying infrastructure. Thus, as soon as users successfully completed the aforementioned tasks the framework solution is operational.

[10] The existence of those policies is no key aspect for the solution to be a success. However, they allow users simplified ways to modify collected identity attributes as part of ongoing data management and maintenance.

[11] As a reason that integration of the well known and broadly used *MIT PGP Public Key Server* is proposed, it is very likely that many SPs already host their public keys at this location.

[12] For simplicity reasons, programmers could also integrate and automate this registration into the initial system configuration.

# CHAPTER 5

# Discussion

Chapter 5 interprets the framework results that are presented in Chapter 4. It derives discussion criteria based on the previous chapters and applies them to current practices and systems in the working area and to the framework solution itself. This shows advantages and limitations of the work.

The first section starts with establishing seven discussion criteria. It clarifies their origin and how they are applied to analyze the systems. These criteria cover all project relevant aspects regarding identity, privacy and mobility.

In the next section those criteria are applied to current practices and systems in order to show related advantages and disadvantages. Internalization of them helps to identify challenges that the framework needs to face.

Section 5.3 then applies the same criteria to the framework solution. Thereby, main focus is placed on the three most important aspects *Security*, *User Trust* and *Privacy*. Personal opinions also show limitations and potential improvements of the proposed system.

The final section supplements the discussion. It briefly describes the current project status and reviews research questions and user scenarios. Last but not least, a presentation of improvements to other systems visualizes the framework's advantages and user benefits.

## 5.1 Discussion Criteria

According to the development strategy of Subsection 1.4.5, this project concludes with the solution design and development. Thereby, even though the proposed theoretical framework was validated to widest possible extent, successful deployment still requires extended testing after prototyping.

In order to simplify discussion against different sets of requirements, criteria are derived from the project objectives (Subsection 1.2.2), the user scenario questions (Subsection 3.1.2), the functional requirements (Table 3.2), the framework specifications and the PM Design Guidelines (both Table 3.1). They are supplemented by further important criteria that were not covered during project execution so far (e.g. energy consumption). Still focusing on the mobile area, they allow verification of all project relevant aspects. In the following two sections these criteria are applied to analyze current practices and systems in the treated working area and the framework itself. The last section compares both verifications and finally discusses the advantages and disadvantages of the proposed solution.

The next paragraphs list the used criteria, whose first four items mainly reflect the Basic Framework Pillars within IM. The others are then derived from the aforementioned aspects, the project context and the treated working area. It is important, that the presented results show theoretical considerations. The practical verification needs to be complemented during and after prototyping.

**Security**
Analysis to show to what degree data is protected on the devices, during transmission and in cases of device lost. Does the proposed security approach sustain continuous user anonymity? What is the impact of applied cryptography on the processing capabilities? Are the default security settings appropriately chosen for various contexts and adjustable by users?

**User Trust**
Two important questions to face are, what kind of user trust relations are required? And how are users supported in establishing those trust relations?

**Cost Efficiency**
Discussion on which one time expenses result from roll out and which permanent costs from running, managing and maintaining. Is there a balance between costs and the system's added value?

**Ease of Use**
Even though it is very hard to theoretically verify, the systems are analyzed regarding ease and naturalness of use. The discussion is evaluated towards basic users which in the project's context means users with fundamental knowledge on how to browse the internet and manage simple interface dialogs on mobile phones. It needs to be identified if system and transactions are permanently transparent. Do users retain responsibility and control throughout entire transactions?

**Privacy**
Twelve PM Design Requirements were specified during project execution (Table 3.1). They are used to theoretically evaluate privacy awareness of the observed systems. The question is, what is the degree of performance concerning those requirements?

**Mobility and Flexibility**
The addressed systems are built for roaming users. Thus, it is required to analyze aspects regarding dependencies to other systems units or resources. To what extent are users obliged to apply preconfigured and fixed pseudonyms? Are there any restrictions or limitations regarding the field of operation?

**Energy Consumption**
Besides processing capabilities (addressed in the first criteria) battery power is limited on mobile phones. Thus, the systems are theoretically analyzed regarding expected energy consumption.

## 5.2 Current Practices and Systems in the Working Area

In this section current practices and systems in the working area are discussed by matching all criteria from Section 5.1. Thereby, main input is derived from the reference systems that were indicated during design and development. However, the analysis also takes into account general perceptions that have been gathered during background research. It needs to be noted that theoretical evaluation without comparing any prototypes is challenging and mostly only based on public documentations.

**Security**
Currently available systems follow a variety of security concepts. Unfortunately, applied practices are rarely published, but basically most solutions implement public key cryptography. Thereby, one system makes use of session keys that are exchanged before any transaction is initiated. However, this approach obviously prevents anonymity. Regarding secure data storages there is also almost no information available. While one system outsources identity data to central registrars and thus circumvents encryption on mobile phones, another solution securely stores encryption keys on SIM cards. This approach may be susceptible for brute force on the PIN and the registrar is highly insecure if it requires no identification that could be blocked. However, no analyzed infrastructure considers risks of device lost.

**User Trust**
Actually, none of the investigated systems focuses on minimizing user trust relations. One solution even goes so far to relocate privacy-sensitive identity data entirely to central registrars what makes it very hard to achieve appropriate user trust. Other systems propose using CAs but do not address trustiness aspects. It was generally reviewed that no solution satisfactory supports users in establishing necessary trust relations.

**Cost Efficiency**
The cost efficiency criterion is almost impossible to apply to current systems. It is believed that especially the solution that requires modified SIM cards is highly expensive in it's development. Even though running costs of all systems are hardly imaginable it can be stated that no efficient balance between costs and added values exist. This is because of various limitations in the operational field and extensive, individual adjustments and modifications on SP sides that are not based on standardized technologies.

**Ease of Use**
One analyzed solution stood out with it's user interface. Simple ways to interact and provide user consents are well designed. It includes brief notifications about currently disclosed pseudonyms to make the system transparent. Unfortunately, complex program sequences hinder optimal handling by basic users. A second system that made a positive impression integrates identity confirmations with Short Message Service (SMS). These widely known SMSs do not make the system very clearly arranged but provide easy interaction for many users. However, the third approach outsources necessary policy generations to personal computers with bigger screen sizes. This unnecessarily complicates overall system use. As far as it could be determined from all solutions, users are not able to retain complete responsibility and control throughout entire transactions. As a conclusion, all evaluations are just impressions from the documentations, no prototypes were applied.

**Privacy**
Obviously no solution fulfills all specified privacy requirements. But as a reason that those are particularly chosen for the project context, this does not mean that other systems do not work privacy-aware. Main aspects that attracted attention in this context are that often fixed pseudonyms and static certificates prevent the principles of data minimization, anonymity and pseudonymity. Moreover, the solutions do not provide ways to easily access and modify collected data on SP side and to prevent unlinkability and untraceability. Addressing positive aspects, logging mechanisms are rarely (and partly) implemented. And a few user interfaces even allow solving of identity conflicts (e.g. disclosed less data) and providing of active user consents.

**Mobility and Flexibility**
Even though the analyzed systems are designed for the mobile area, they prevent full flexibility and mobility by limiting underlying communication infrastructures (e.g. to NFC), available services and uses of individual pseudonyms. Especially the addressed policy generator on personal computers entirely hinders independent uses while roaming.

**Energy Consumption**
Energy consumption can not been determined without using prototypes. The only aspect that was possible to derive from documentations belongs to the solution that makes use of SIM cards. Here, those cards execute all required encryption processes very fast and with low energy consumption. However, it was not possible to evaluate the encryption's impact on processing power and energy consumption in the other solutions.

## 5.3 Framework Solution

This section discusses the framework solution against the criteria of Section 5.1. The three most important aspects to deal with are *Security*, *User Trust* and *Privacy*. This is the reason why the related discussions are explored in more depth than the remaining ones.

**Security**
Analyzing framework aspects related to security there are mainly two areas to focus on, storage and transmission protection. Data storages are designed in a decentralized approach (Subsection 2.1.4)[1]. Directly integrated into mobile phones, data is protected with symmetric key cryptography. This way is chosen because it is widely proven to be fast and to require low processing power and thus work smoothly on mobile phones with limited capabilities. To provide high security even in cases of loss, keys are encrypted with KEKs and hosted on an internal key server that prevents unauthorized access by requiring two factor authentication. Thereby, the required factors are *existence of a mobile phone with matching KEK* and the *permit to request keys from the server*. Moreover, all encryption keys are only kept alive during program runtime and deleted from mobile phones afterwards. Even though this implementation design is stable and well proven in current systems there are two drawbacks to face within this project. Firstly, the necessary encryption functionality on Android platforms is not existent so far, but as mentioned in Subsection 3.3.4 according to large demands it is most likely to be available soon. Secondly, there is no standardized solution for the internal key server. Thus, this unit represents a non-tested modification of the validated public key server concept. However, it should be easy to technically realize, because the main difference is that keys are rather stored encrypted than in plaintext.

The design for protection of anonymous transmissions was inspired by the widely accepted TOR approach. On the same basis as applied to storage security it is also based on symmetric key cryptography. However, it combines this concept with asymmetric encryption in order to securely agree on session keys. This has the main advantage that it only requires SPs to publish keys, users can permanently stay anonymous. Furthermore, the decision to choose the MIT PGP Public Key Server guarantees a very secure, stable and extensively tested solution. To achieve even higher security in the designed infrastructure, all data messages (except from browsing activities) are split

[1] Table 2.1 showed five particular disadvantages of decentralized over centralized storages. As a conclusion, except of the backup that has not been treated in this project, all items have been successfully eliminated.

into packets, whereby the default amount enables a good balance between speed, security and privacy. However, users are still allowed to make adjustments in every transaction[2]. Applying splitting also during session key agreements makes it almost impossible for attackers to modify corresponding keys. As soon as SPs recognize data that differs from other key packets, malicious information is either rerequested or entire communications are rejected. And even if faked keys would have been agreed upon there is no risk for disclosed user data to be read by attackers. This is, because in this network mobile phones are not aware about any key changes and thus still encrypt data packets with session keys they think they agreed upon with the SP. An analysis showed that approaches of splitting together with the concept of anonymous transmissions are already tested and validated by other solutions. The extension by a central registration unit that is responsible for address management even allows higher degrees of anonymity in the proposed network infrastructure than it is provided in other systems. This is based on the fact that no interstation is aware of the entire infrastructure.

Like for storage protection, the anonymous network infrastructure comes with some challenges. Thus, it so far requires SPs to implement the proposed security protocol and mobile phones to permanently run an appropriate service. Without this, the intended peer-to-peer approach is not realizable. This service unfortunately results in energy consumption but this is discussed separately. A further challenge lies in the registration unit that probably represents the weakest system in the entire infrastructure. If an attacker is able to provide faked addresses the whole concept can become very harmful. Therefore, protection of the unit itself on high levels is crucial. Furthermore, potential cooperations between SPs and the unit need to be prevented. This is, because there may be a chance to determine sending mobile phones based on the fact that they are the only ones that request more than one interstation address for data transmissions.

Finally, it needs to be stated why mobile phones where chosen as interstations rather than a couple of fixed system units. One reason is, that parallel oriented fixed units always know the sending mobile phone because this is the one they receive data from and deliver data to. A randomly changing cascade could be applicable but as soon as more interstations than fixed units are available, they provide much wider variations in the applicable communication paths. Moreover, in the proposed process sequence every additional interstation increases anonymity by many times, whereas fixed units only provide a static maximum for those levels. Last but not least, it may be too expensive to integrate large numbers of fixed units, but low numbers provide unsatisfying anonymity and also increase risks of cooperations between units and SPs. And this unfortunately could lead to anonymity breaks.

**User Trust**

IM and PM functionalities are not realizable without trust relations. Like in all information systems, every unit requires some kind of belief. Even though the framework tries to keep those relations at minimum, it is not possible to entirely avoid them. In this context, standardization and certification of affected systems can support users in establishing all necessary trust relations. Thereby, certificates would guarantee that the units work according to particular standards. One way for this is the so-called *Certification of Secure Infrastructures for IT Systems* by TUViT[3].

[2] Generally it can be said, the higher the number of packets the higher anonymity and security and the higher risks of packet loss during transmission.

[3] TUViT is a Trust Provider for industry and official authorities that mainly operates in German speaking countries. For more information see `http://www.tuvit.de/english/Infrastructure.asp`.

Another possibility is the commonly and worldwide used *Information Technology Infrastructure Library (ITIL)*[4].

Generally, no trust is required for central storages. With databases on mobile phones, users themselves are responsible for appropriate handling. This approach is proven to be more accepted than establishing user trust into *invisible* central storages. For trust reasons, the proposed architecture also circumvents central identity providers (see Figure 2.1) and directly integrates related tasks into the mobile phones. However, trust in the following system units is still required.

**External Key Server** - To assist users in the best possible way of establishing this trust relation, implementation of a well known and verified server (MIT PGP Public Key Server) was proposed.

**Internal Key Server** - There is no specific way to support users in establishing this trust relation, except for a combination of the server with the following registration unit. However, whether one combined system is more likely accepted by users than two separate units needs to be evaluated during prototyping.

**Registration Unit** - This particular trust relation is necessary, unless users waive anonymity and rather want to directly communicate with Service Providers. As said for the internal key server, a higher degree of trust may be achieved by combining both systems.

**Interstations** - The proposed peer-to-peer approach proved to be successful in file sharing purposes. Furthermore, in order to increase user trust, implementation of individual blacklists is planned that allow users to mark particular interstations as malicious. The goal is to increase well-feeling on user side. The same as for the registration unit, this trust is only required when using the anonymous network.

**Security Protocol** - Users need to trust a security protocol as middleware on SP sides. A transformation of the design into an open standard could be very helpful for assisting users and also SPs in trust establishment[5], because open standards are permanently tested, reviewed and verified.

**Data Management Practice Policies** - By now, there is no enforcement mechanism integrated for SP compliance with published data management practices. However, if users do not want to work in this *uncertainty*, they can use the provided network without privacy technology. But then, anonymity and user support in disclosing identity data are limited. Therefore, an appropriate functionality is planned to be added to the solution soon[6].

**Cost Efficiency**

Usage of existing underlying hardware and open source solutions results in low-cost framework development. Furthermore, the used mobile phone platform[7] also makes inexpensive rollout possible. Thus, the only costs are related to manpower for programming, testing and deployment. However, even this is limited, based on a solution that is already entirely designed and prepared for handover.

[4] ITIL provides consistent and comprehensive documentation of best practices for information technology service management. For more information see `http://www.itil-officialsite.com/`.

[5] One way for standardization is presented by the Organization for the Advancement of Structured Information Standards (OASIS). The vision of OASIS is to drive development, convergence and adaption of open standards for the global information society. For more information see `http://www.oasis-open.org/`.

[6] For this aspect review also the upcoming discussion within privacy.

[7] The Android platform is open; new applications do not require any certification like in comparable systems.

Addressing running expenses, costs for network connectivity in order to browse and use services are just the same as carrying out those tasks without framework support. However, all mobile phones in the anonymous network are required to permanently run a peer-to-peer service. Consulted only from time to time, in standby mode it does not provide any or at least very limited network traffic. Moreover, data packets are small and thus not that costly to transfer. The challenge rather lies in forwarding non-split browsing data in an inexpensive way. An improvement to the designed systems in relation to lower bandwidth usage might be to also split such kind of data. But unfortunately this will slow down related tasks because of longer transmission times and risks of packet loss. However, all running costs are balanced due to simplification and probably saving time[8] for privacy-aware online identifications; especially in the current time of increasing flat rate contracts.

**Ease of Use**
Coverage of usability aspects belongs to the limitations of this project. Nevertheless, all activities are based on a simplified browser, so that users with basic knowledge in this area are addressed. This browser automatically starts the required interfaces that allow privacy-sensitive handling of identity data. Besides this, the user tool enables management and maintenance of this data to be performed directly on mobile phones. Adjustable frequencies of notification messages, together with a permanently visible progress bar and an emergency exit button make system, processes and transactions transparent and user-centered. Furthermore, the user tool automates as much tasks as possible, while requests for active user consents permanently ensure to retain responsibility and control.

**Privacy**
It is very useful to base privacy related development on widely established design guidelines. This helps in building well structured and coherent overall concepts and simplifies reviews by persons not directly involved in the development process. With this starting point, framework specific requirements were derived from commonly used privacy specifications (Table 3.1). An analysis of their fulfillment is applied to verify and validate the proposed solution. Anticipating the following list, the conclusion is that all requirements are satisfied as far as possible. The designed framework guarantees handling of identity data on high privacy levels. However, there is still space for improvements, especially for PDR 2, 7 and 12, as shown shortly. Further development is therefore intended to be carried out in future.

**PDR 1. Data Minimization** - Based on standardized P3P policies[9], only required identity attributes are automatically disclosed as transaction dependent pseudonyms.

**PDR 2. Ease of Access and Revocation** - Modifications in the local databases lead to user notification about affected SPs and a presentation of human readable policies that specify particular ways of access. Besides this, log files allow reviews of disclosed attributes and applied data management practices. A potential improvement of PDR 2 is a mechanism to automatically update data on SP sides.

**PDR 3. Security** - Storage and transmission protection allow data confidentiality and integrity.

[8] Time measures need to be established and validated during prototyping.
[9] The framework works with a reduced set of P3P vocabulary that reflects all project relevant data items. In cases, users or SPs need to specify more than the included information the user tool needs adjustments.

**PDR 4. Logging** - Transactions and system changes are logged. The corresponding log files allow manual reviews by users and automated analyses by the user tool.

**PDR 5. Pseudonymity and Anonymity** - Pseudonyms integrate context dependent identity attributes and are anonymously transmitted. Users are able to adjust chosen anonymity levels, whose default settings are suitable for all transactions *(fast, anonymous and secure)*.

**PDR 6. Unlinkability** - The user tool automatically notifies users in case of data concatenation risks.

**PDR 7. Untraceability** - the user tool automatically notifies users in case of position tracking risks. The tool also guarantees that no locational data is disclosed unnoticed. Untraceability can be improved by integrating additional mechanisms, like for example the *k-anonymity* concept[10].

**PDR 8. User Centering and Transparency** - The user tool is oriented towards users. Various mechanisms inform about current steps and make system, processes and transactions transparent. The tool also allows termination of activities and transactions at any time.

**PDR 9. User Consent** - All critical processes require active user consent. However, experienced users are able to adjust the frequency of related requests.

**PDR 10. System Integration** - All privacy protection functionalities are directly integrated into the user tool; no additional implementation steps are necessary.

**PDR 11. User Empowering** - Privacy levels are automatically analyzed and require confirmation by users. User anonymity in transmissions can be adjusted by changing the amount of packets in that data messages are split.

**PDR 12. Multilateral Security** - Necessary trust relations are kept at minimum. Related improvements are based on certifying trustable system units in order to support trust establishment[11].

However, even if all these requirements are met, the framework still shows open privacy issues. One is, that the entire approach is based on the P3P technology. This means, if SPs do not publish related policies, neither user support can be provided nor privacy be guaranteed. On the other side, when SPs nonetheless integrate the proposed framework security protocol, users are able to carry out transactions through the anonymous network. A further challenge that was already addressed previously is that no enforcement mechanisms are integrated to guarantee that SPs comply with publish practices. Improvement in this context could be made with so-called *sticky policies*. The idea behind them was established during the PRIME project[12] [57]. Comparable to DRM technology, those policies allow to enforce data practices after disclosure. A similar feature will be included in a later framework version.

As a conclusion, the initial privacy issues that were stated during background research in Subsection 2.2.2 are successfully mitigated by the proposed solution.

[10] A station is *k-anonymous* if it's locational data is indistinguishable from that of at least *k-1* other stations.

[11] For more information about user trust and related improvements it is referred to the corresponding discussion.

[12] The development goal of the PRIME project was a privacy enhancing IDMS prototype. The original work is ended but the basic concept is carried on as *PrimeLife*. For more information see `https://www.prime-project.eu/` and `http://www.primelife.eu/`. An equal approach is also shown within the PRiMMA discussion in Subsection 1.3.2

**Mobility and Flexibility**
Abdication of central storages and modified SIM cards makes the user tool independently and free to install on mobile phones. Participation in the anonymous network then requires user to register their devices at the specified DNS service. While the decentralized storage approach also allows users to manage and maintain identity data without internet connectivity, WLAN or UMTS communication is required for online transactions. This is the reason, why particular system units are placed in the network subsystem that is opened to the internet, as shown in Figure 3.26. As a conclusion, roaming users are able to apply the framework to all online services that require identification, as long as the SP made appropriate configurations, as show in Subsection 4.4.2. In this way, the overall solution aspires a large operational area and the use of context dependent and individual rather than preconfigured and static pseudonyms builds up the basis for high flexibility.

**Energy Consumption**
Addressing energy consumption a very vital aspect relates to the anonymous network. As stated previously, participation requires a permanently running peer-to-peer service. Even in standby mode this service consumes battery power while waiting for requests. But unfortunately, none of the analyzed alternatives was able to achieve user anonymity on comparable high levels. Under those aspects and in order to reduce energy consumption for encryption, symmetric key cryptography was chosen because it proved to be sufficient for the mobile area with limited processing power. Furthermore, for security reasons it was decided that data on mobile phones is not permanently readable and only decrypted on requests. While this concept provides higher security, frequent program executions lead to increased energy consumption. But this was accepted since security aspects are assumed to have priority in this project. Last but not least, to counteract the overall consumption issue, unnecessary gimmicks were avoided in the interfaces and it was tried to integrate as many static elements as possible. As a conclusion, accurate measures of actual energy consumption need to be established and evaluated during prototyping. This is also the development point at which improvements can best be identified.

## 5.4 Solution Summary

This section concludes Chapter 5 and introduces the project conclusion. The first part supplements the previous discussion by briefly describing the project status and reviewing research questions and user scenarios. The second part works out the solution's improvements to other systems and visualizes related advantages that show achieved user benefits.

### 5.4.1 Solution Verification

According to the project development strategy (Subsection 1.4.5), this work concludes with the design and development. Therefore, the chosen discussion criteria helped to verify the solution. The results ease project handover to programmers because the aspect that this project followed commonly accepted design guidelines results in a high probability that the presented framework is valid. Programmers themselves are then responsible for practical verification based on prototypes.

With reference to the first three research questions, they are answered in Chapters 2 to 4. Now, that the framework discussion is established, it is also clarified to what extent mobile devices can be used for IM and PM purposes while roaming (research question four). In this context, the presentation

of challenges and issues related to the different discussion criteria showed the solution's limitations. Moreover, since all research questions are solved, the connected project objectives of Subsection 1.2.2 are successfully fulfilled, too.

Briefly reviewing the two typical user scenarios of Subsection 3.1.2, the framework covers related identity and privacy problems. Thus, the solution helps the fictional characters Carol and Leon to automatically fill in online forms, it warns Carol that her data can be forwarded to marketing companies and Leon about risks of user data concatenation. It also aids Carol remembering and submitting login data. The following subsection finally shows the solution's improvements to other systems and further focuses on it's advantages and user benefits.

### 5.4.2 Improvements to Current Systems

As a summary of the framework's main improvements to other systems the following project contribution can be stated. It shows that the solution complies with the contribution that is defined in Section 1.5. As required, the framework allows users to work with different levels of privacy, anonymity, accountability and confidentiality. Thereby, the last two aspects are mostly covered by a voluntary participation in the anonymous network and default privacy and security settings that are freely adjustable by users.

**Storage** - Privacy-sensitive data is stored in a protected database on mobile phones rather than in centralized systems. Furthermore, there is no need to install additional storage hardware (e.g. SIM cards).

**Pseudonyms** - The proposed entirely different overall concept applies individual and context dependent pseudonyms rather than preconfigured and static ones, like used in all analyzed solutions.

**Risk Analysis** - An entire novelty to other systems is the designed functionality that allows to identify privacy risks and issues before data is disclosed.

**User Trust** - In contrast to comparable solutions, the framework son working with the lowest amount of user trust as possible. Consequently, users are highly supported in establishing all needed trust relations (e.g. by integration of a broadly verified external key server).

**User Tool** - The user tool covers a novel option to control the entire data life cycle. This starts with storing data on mobile phones and continues with ongoing data management and maintenance on SP side. Especially the second step is not integrated in any other identified system so far.

**Operational Field** - The framework does not restrict its use to particular services. It is applicable to all online identifications, as long as the SP publishes data management policies and integrates the designed security protocol. Furthermore, the design allows users to anonymously disclose location based data in order to receive corresponding services. A similar mechanism could not be analyzed in comparable solutions.

# CHAPTER 6

# Conclusion

The first section of Chapter 6 reviews the motivation and background for the entire project. Thereby, it reemphasizes on the research questions and the applied development strategy and also briefly picks up the project handover to programmers.

In the second section the project results and the most important basic and technical conclusions are shown.

The third section sums up the key contributions to knowledge that this work has established. It also shows the achieved implications for users and SPs. It finally briefly identifies advantages for those SPs that decide to participate in the framework solution.

In the last section future work is discussed. An application roadmap presents the project handover to programmers and the related visualization demonstrates subsequent tasks that are required in order to successfully roll out the framework. Thoughts about planned improvements and recommendations for further research in the project field round up this chapter and the entire thesis.

## 6.1 Motivation and Problem

The research and the subsequent design and development process in this project revealed several interesting new aspects. Resulting from increasing user demands for flexible and privacy-sensitive support in online identifications, the project's main goal was the development of a framework for IM and PM on mobile devices. The solution was conceived to be an improvement of currently available approaches with particular regard to roaming users. Because of this, focus was especially placed on flexibility and mobility aspects. According to all these high-level specifications, the following four research questions and their corresponding objectives were asked.

1. What are the main advantages, disadvantages and challenges when using IM and PM on mobile devices?
2. What are the requirements for the framework parts - the procedural method, privacy and security protocols and user tool? And do corresponding solutions exist that can be adapted or is it necessary to develop them?
3. What kind of communication infrastructure and technology is needed and what are the requirements for SPs to enable application of the framework solution?
4. To what extent does the framework solution enable roaming users with proof of identity while preserving high privacy levels?

To find appropriate answers, a development strategy was applied that consists of six working steps, as seen in Figure 1.1. Due to limited time, the first three steps are the ones covered by this project (background research, requirements and specifications, design and development). However, the proposed solution is designed and thus prepared to such detail that programmers can directly start coding and testing.

In the last subsequent development step the solution will be deployed. Therefore, a detailed discussion of related tasks is presented in Section 4.4 and also as part of the future work in Section 6.4. That section shows an application roadmap for subsequent development steps and specifies responsibilities for the solution rollout.

According to the chosen development strategy, the background research established an insight into current work in the treated field. Thus it allowed identification of advantages and limitations of today's approaches. These findings were used to establish framework requirements and specifications in order to improve analyzed challenges. At this stage it can be stated that the designed solution covers all defined requirements and specifications. It also fulfills all objectives and answers the four research questions. The suggested user tool, as part of the framework, supports roaming users in applying identity data in online scenarios and guarantees privacy-sensitive handling of related information. To complete this project, the following sections review results, contributions and conclusions and finally present thoughts about further work.

## 6.2 Results and Conclusions

The main result of this work is that the project goal and the related objects are fully met: A framework for IM and PM on mobile devices has been developed as specified. Based on the assumptions made in Section 1.4 it is believed that the designed solution is correct.

In Subsections 6.2.1 and 6.2.2 two lists provide the most important basic and technically oriented conclusions. They show that the limitations of current approaches (Section 1.3) have been eliminated. As soon as the proposed framework is coded, tested and deployed, it allows users simplified and privacy-sensitive disclosure and handling of digital identities in online scenarios.

### 6.2.1 Basic Conclusions

1. The problem stated in Section 1.2 has been solved and the main project goal of Subsection 1.2.1 achieved. As shown in Chapter 3, a framework for IM and PM on mobile devices has been designed.
2. All non-functional requirements of Subsection 1.2.3 have been fulfilled during development.
3. Main advantages, disadvantages and challenges when using IM and PM functionalities on mobile devices have been identified in Subsections 2.1.4 and 2.1.5.
4. Existing solutions for the framework parts *procedural method*, *privacy and security protocols* and *user tool* have been identified and adapted in places, as discussed in Sections 3.2 and 3.3.
5. The necessary communication infrastructure and the underlying mobile phone technology have been specified in Subsections 3.2.1 and 3.4.3.

6. Requirements for SPs have been defined in Subsection 3.4.3, that enable them to adapt services for being used within the framework solution.
7. The extend to that the framework enables roaming users with identity proofs while preserving high privacy levels has been stated in Chapter 5.

### 6.2.2 Technical Conclusions

1. Requirements for the framework parts *procedural method*, *privacy and security protocols* and *user tool* have been defined in Section 3.2.
2. Framework aspects within privacy have been developed according to the PM Design Guidelines of Subsection 2.2.4.
3. The Privacy Design Requirements of Subsection 2.2.4 have been fulfilled as best as possible.
4. All functional requirements of Subsection 3.2.5 have been met during development.
5. Framework aspects within privacy have been designed to work on the basis of the P3P technology.
6. Encryption of data on mobile phones has been realized with symmetric key cryptography, whereby keys are stored encrypted on an internal key server in the anonymous network.
7. An artificial network has been proposed that enables secure and anonymous communication.
8. Encryption of data transmissions has been designed to work with session key technology in that SPs publish corresponding keys available on an external key server on the internet.
9. The user tool has been developed to automatically carry out as much tasks as possible, whereby users permanently retain responsibility and control.

## 6.3 Summary of Contributions and Implications

This section provides an overview of the contributions to knowledge and shows the implications for roaming users and SPs that this master's thesis has made.

The first list briefly sums up the key contribution, whereby all made claims have been substantiated throughout the report. Furthermore, the presented items are closely connected to the improvements to current systems of Subsection 5.4.2. The section then presents the user implications of the project solution. Finally implications and advantages for SPs that decide to participate in the framework solution are shown.

### 6.3.1 Project Contributions

1. An entire novelty of this project is the automated detection of potential privacy risks and issues before data disclosure.
2. The framework works with a new concept of individual and context dependent, rather than preconfigured and static pseudonyms.
3. The solution proposes a new approach of a peer-to-peer network that basically follows the TOR concept and the design of [34]. Particular extensions and improvements allow higher anonymity and require less user trust in data transmissions than comparable solutions.
4. The project demonstrates the first use of user-centered anonymity and security level adjustments for data transmissions. It allows to choose the amount of packets, data messages are split into. This feature is not included in any other analyzed system so far.
5. The framework proposes a new concept to enable anonymous disclosures of positional data with simultaneous participation in location based services.
6. In contrast to comparable solutions, the system design focuses much more on user trust. It works with a low amount of trust relations and highly supports users in establishing these (e.g. by implementing a broadly verified external key server).
7. The framework proposes a new approach to outsource encryption keys for data on mobile phones to a modified key server. The developed infrastructure thereby allows to prevent access in cases of device loss, what makes privacy-sensitive data unreadable (temporary and permanently).
8. The designed user tool covers the entire life cycle of identity data. Especially post disclosure data maintenance on SP side is not integrated in any other analyzed system so far.
9. The user tool allows to keep track of disclosures and system changes by creating log files. No such mechanism could be found in the majority of the analyzed systems.
10. The framework is much more flexible than comparable approaches presented in this report. It can be applied to any online identification, as long as SPs prepare their services accordingly. Thereby, the used vocabulary for requesting identity attributes is freely extensible.

### 6.3.2 Roaming User Implications

1. Identity attributes on the mobile phones allow high flexibility and mobility for roaming users.
2. Automated analyses of data management policies ease understanding of applied practices and enable simplified agreements based on individual user demands.
3. The designed framework simplifies online identification and most probably reduces processing time on user side[1].
4. Automated dialogs and transparent interfaces allow quick familiarization with the working environment and thus highly support users in the learning process of using the new system.
5. Freely adjustable notification messages enable sensitization of both, beginners and experienced users for identity related privacy risks and issues in online transactions.

[1] Whether processing time is really reduced or not needs to be evaluated during prototyping.

### 6.3.3 Service Provider Implications and Advantages

In the previous subsections the project contributions and roaming user implications have been stated. However, the entire solution is strongly dependent on SP participation. If one of them does not integrate the security protocol, publish data management policies and provide public keys, the framework is not applicable to it's broadest extent when requesting services from that particular SP[2]. In this context it is important to note that the solution does not require any extra effort from SPs during transactions, because all tasks are automatically handled by the framework security protocol that is placed as a middleware on their side. It transforms requests and responses in that manner that SPs can handle transactions in the usual way. Based on these facts, the following key implications and advantages for SPs are in given.

1. The provided possibility for anonymous exchanges of privacy-sensitive data on high security levels most likely increases user trust in the SP.
2. The framework security protocol integrates a mechanism for automatically initiated, encrypted responses. The key point is, that this feature is always available, regardless of the SPs infrastructure.
3. The proposed solution guarantees complete awareness of applied data management practices on user side. Even if this is probably the same with written GTCs, it is widely known that most users instinctively accept GTC statements without reading them. However, the framework allows automated analyses of policy specifications and thus strongly supports user awareness.
4. The designed system increases the probability that collected data on SP side is permanently valid and updated. Without framework support, users could tend to miss modification or even leave out a couple of affected SPs.

## 6.4 Future Work

This final section shares thoughts about how the presented thesis can be applied as starting point for future research and work. Therefore, an application roadmap visualizes outstanding steps of the applied development strategy and shows subsequent processes on the way to the framework rollout. In the second part, planned improvements on the project solution and recommendations for future research are presented.

### 6.4.1 Application Roadmap

The presented project has been carried out according to the development strategy of Figure 1.1, whereby only the first three steps are totally covered. This subsection comes in addition to the remarks of Subsection 1.4.5 and the detailed overview of subsequent development steps of Section 4.4. It presents an application roadmap that visualizes following tasks that are required on the way

[2] In service requests without policies the user tool displays online forms and notifies about disabled functionalities. Furthermore, encryption is dependent on SP configuration; framework mechanisms do not apply. Last but not least, data is not split and anonymously sent and received as a single packet. The case, that SPs publish policies but do not integrate the security protocol is not covered by the current framework version. Here, split messages are unreadable for SPs.

to the framework rollout. Figure 6.1 thus shows the three participants *Programmers and System Architects*, *Users* and *Service Providers*. It clarifies responsibilities and defines related tasks for each of the three outstanding steps.

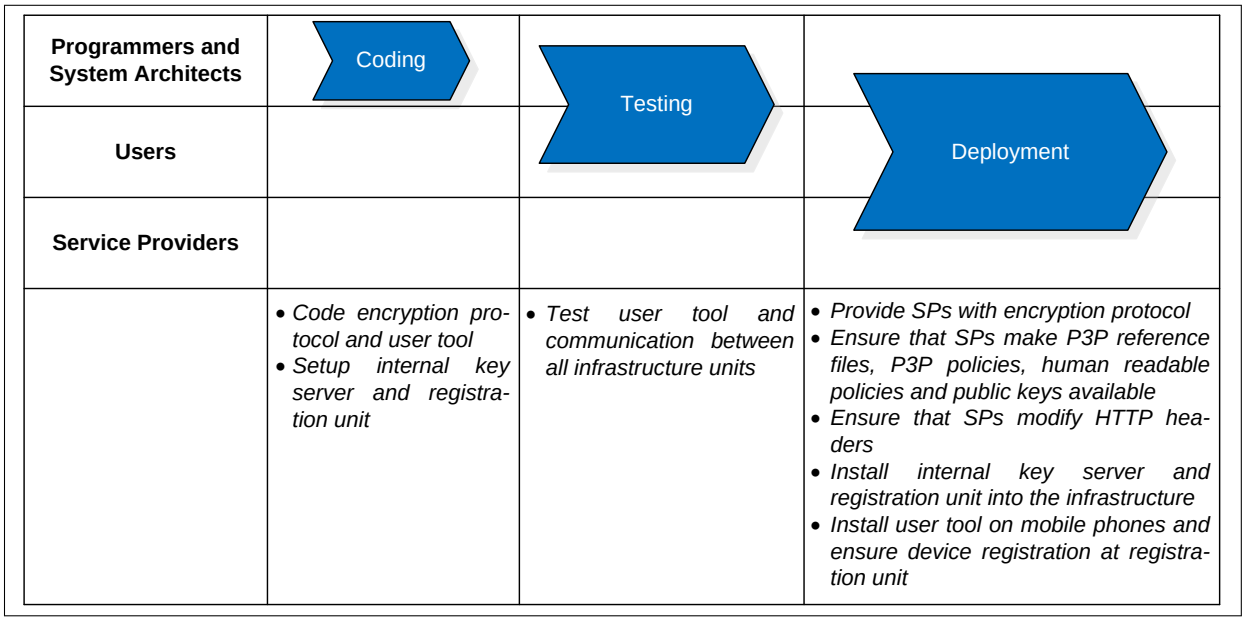



Figure 6.1: Framework application roadmap for the final rollout

### 6.4.2 Improvements and Recommendations

This subsection shows the improvements that are already planned for the current framework solution and gives recommendations for further research.

**Planned Improvements**

1. Integration of individual blacklists that allow users to ignore particular interstations in data transmissions.
2. Realization of an enforcement method that guarantees compliance with published data management practice policies on SP side.
3. Implementation of a functionality to automatically update modified identity attributes on SP side (to the widest extent possible).
4. Execution of certification processes for involved system units in order to achieve higher user trust.
5. Perhaps: Improvement of the untraceability aspect by integrating the k-anonymity concept.

### Research Recommendations

1. The peer-to-peer service on the mobile phones requires a permanent internet connection and thus probably constitutes the major proportion of the framework's energy and bandwidth consumption. It should be identified if there are ways to generally reduce necessary processing power and bandwidth usage.
2. It should be studied if variations in the used communication channels can reduce bandwidth consumption and thus internet connection fees on interstation side. For example, by using NFC whenever available, only delivering interstations should require communication over the internet. However, the potential extent of application needs to be identified.
3. One should analyze whether transformation of the designed security protocol into an open standard solution is possible or not, because this would most likely increase acceptance on SP side.
4. It can be the case, that users apply the same identity attributes and request equal services also from their personal computers. Therefore, it could be interesting to study, if it is generally possible to synchronize the user tool with stationary computers. It then needs to be identified to what extent the anonymous network still can be applied, when working on the computer rather than on a mobile phone.

# Acronyms

| | |
|---|---|
| AICPA | American Institute of Certified Public Accountants |
| CA | Certification Authority |
| DNS | Domain Name System |
| DRM | Digital Rights Management |
| FIPPs | Fair Information Practice Principles |
| GPS | Global Positioning System |
| GTC | General Terms and Conditions |
| HTTP | Hypertext Transfer Protocol |
| IDMS | Identity Management System |
| IM | Identity Management |
| IP | Internet Protocol |
| IPv6 | Internet Protocol Version 6 |
| ITIL | Information Technology Infrastructure Library |
| KEK | Key Encryption Key |
| MAC | Media Access Control |
| NFC | Near Field Communication |
| OASIS | Organization for the Advancement of Structured Information Standards |
| OSI | Open Systems Interconnection |
| P3P | Platform for Privacy Preferences Project |
| PDA | Personal Digital Assistant |
| PET | Privacy Enhancing Technologies |
| PIN | Personal Identification Number |
| PM | Privacy Management |

| | |
|---|---|
| PRiMMA | Privacy Rights Management for Mobile Applications |
| SIM | Subscriber Identity Module |
| SMS | Short Message Service |
| SP | Service Provider |
| TCP/IP | Transmission Control Protocol / Internet Protocol |
| UML | Unified Modeling Language |
| UMTS | Universal Mobile Telecommunications System |
| URI | Uniform Resource Identifier |
| W3C | World Wide Web Consortium |
| WLAN | Wireless Local Area Network |
| XML | eXtensible Markup Language |

# Bibliography


[1] OpenID Foundation. *OpenID Authentication 1.1.* May 2006. URL: http://openid.net/specs/openid-authentication-1_1.html (visited on 03/13/2010).

[2] Microsoft Corporation. *Introducing Windows CardSpace.* Apr. 2006. URL: http://msdn.microsoft.com/en-us/library/aa480189.aspx (visited on 03/13/2010).

[3] Eclipse Foundation. *Higgins - Open Source Identity Framework.* 2009. URL: http://www.eclipse.org/higgins/ (visited on 03/13/2010).

[4] Liberty Alliance. *Liberty Alliance Project.* May 2009. URL: http://www.projectliberty.org/ (visited on 03/13/2010).

[5] S. Acharya. *Worldwide mobile cellular subscribers to reach 4 billion mark late 2008.* International Telecommunication Union. Sept. 2008. URL: http://www.itu.int/newsroom/press_releases/2008/29.html (visited on 03/17/2010).

[6] S. Wohlgemuth et al. "Sicherheit und Benutzbarkeit durch Identitätsmanagement." In: *Aktuelle Trends in der Softwareforschung - Tagungsband zum doIT Software-Forschungstag 2003.* Stuttgart: IRB Verlag, 2004. ISBN: 3-8167-6453-3.

[7] K. Hyppönen. "An Open Mobile Identity Tool: An Architecture for Mobile Identity Management." In: *EuroPKI '08: Proceedings of the 5th European PKI workshop on Public Key Infrastructure.* Berlin, Heidelberg: Springer-Verlag, 2008, pp. 207–222. ISBN: 978-3-540-69484-7. DOI: http://dx.doi.org/10.1007/978-3-540-69485-4_15.

[8] F. Paci et al. "VeryIDX - A Privacy Preserving Digital Identity Management System for Mobile Devices." In: *MDM '09: Proceedings of the 2009 Tenth International Conference on Mobile Data Management: Systems, Services and Middleware.* Washington, DC, USA: IEEE Computer Society, 2009, pp. 367–368. ISBN: 978-0-7695-3650-7. DOI: http://dx.doi.org/10.1109/MDM.2009.55.

[9] K. Hyppönen, M. Hassinen, and E. Trichina. "Pseudonymous Mobile Identity Architecture Based on Government-Supported PKI." In: *Trust '08: Proceedings of the 1st international conference on Trusted Computing and Trust in Information Technologies.* Villach, Austria: Springer-Verlag, 2008, pp. 107–118. ISBN: 978-3-540-68978-2. DOI: http://dx.doi.org/10.1007/978-3-540-68979-9_8.

[10] K. Rannenberg. "Identity management in mobile cellular networks and related applications." In: *Information Security Technical Report* 9.1 (2004), pp. 77 –85. ISSN: 1363-4127. DOI: 10.1016/S1363-4127(04)00017-2.

[11] A. Jøsang, M. Zomai, and S. Suriadi. "Usability and privacy in identity management architectures." In: *ACSW '07: Proceedings of the fifth Australasian symposium on ACSW frontiers.* Ballarat, Australia: Australian Computer Society, Inc., 2007, pp. 143–152. ISBN: 1-920-68285-X.

[12] O. Berthold and M. Köhntopp. "Identity management based on P3P." In: *International workshop on Designing privacy enhancing technologies*. Vol. 2009/2001. Berkeley, California, United States: Springer-Verlag New York, Inc., 2001, pp. 141–160. ISBN: 3-540-41724-9.

[13] L. Cranor et al. *The Platform for Privacy Preferences 1.0 (P3P1.0) Specification*. W3C Working Group. 2002. URL: http://www.w3.org/TR/P3P/ (visited on 05/07/2010).

[14] A. K. Bandara et al. "Privacy rights management for mobile applications." In: *4th Int. Symposium on Usable Privacy and Security, Pittsburgh (July 2008)*. URL: http://cups.cs.cmu.edu/soups/2008/posters/bandara.pdf (visited on 05/07/2010).

[15] M. Fahrmair, W. Sitou, and B. Spanfelner. "Security and privacy rights management for mobile and ubiquitous computing." In: *Workshop on UbiComp Privacy*. 2005, pp. 97–08.

[16] D. Hong, Y. Y. Mingxuan, and V. Shen. "Dynamic privacy management: a plug-in service for the middleware in pervasive computing." In: *MobileHCI '05: Proceedings of the 7th international conference on Human computer interaction with mobile devices & services*. Salzburg, Austria: ACM, 2005, pp. 1–8. ISBN: 1-59593-089-2. DOI: http://doi.acm.org/10.1145/1085777.1085779.

[17] J. Lindqvist and L. Takkinen. "Privacy management for secure mobility." In: *WPES '06: Proceedings of the 5th ACM workshop on Privacy in electronic society*. Alexandria, Virginia, USA: ACM, 2006, pp. 63–66. ISBN: 1-59593-556-8. DOI: http://doi.acm.org/10.1145/1179601.1179612.

[18] G. M. Køien and V. A. Oleshchuk. "Personal privacy in a digital world." In: *Telektronikk* 2 (Feb. 2007), pp. 4–19.

[19] V. Gratzer and D. Naccache. "Cryptography, Law Enforcement, and Mobile Communications." In: *IEEE Security and Privacy* 4.6 (2006), pp. 67–70. ISSN: 1540-7993. DOI: http://dx.doi.org/10.1109/MSP.2006.148.

[20] W. C. Booth, Colomb. G. G., and Williams J. M. *The craft of research*. engl. Third Edition. University of Chicago press, 2008. ISBN: 0-226-06565-0.

[21] S. Andova et al. "A framework for compositional verification of security protocols." In: *Inf. Comput.* 206.2-4 (2008), pp. 425–459. ISSN: 0890-5401. DOI: http://dx.doi.org/10.1016/j.ic.2007.07.002.

[22] M. Boreale and M. G. Buscemi. "A framework for the analysis of security protocols." In: *In CONCUR: 13th International Conference on Concurrency Theory. LNCS*. Springer-Verlag, 2002.

[23] N. A. McEvoy. *e-ID as a public utility*. Consult Hyperion. May 2007. URL: http://digitaldebateblogs.typepad.com/digital_identity/EEMA_McEvoy_Utility_1.pdf (visited on 03/15/2010).

[24] A. Bhargav-Spantzel et al. "User centricity: a taxonomy and open issues." In: *Journal of Computer Security* 15.5 (2007), pp. 493–527.

[25] GSM Association. *Identity Management Framework Document (v. 1.1)*. Jan. 2008. URL: http://www.gsmworld.com/documents/identity_management_framework.pdf (visited on 03/18/2010).

[26] T. A. Johansen, I. Jörstad, and D. van Thanh. "Identity Management in Mobile Ubiquitous Environments." In: *Internet Monitoring and Protection, 2008. ICIMP '08. The Third International Conference on* (2008), pp. 178 –183. DOI: 10.1109/ICIMP.2008.29.

[27] D. Chaum. "Security without identification: Transaction systems to make big brother obsolete." In: *Communications of the ACM* 28.10 (1985), pp. 1030–1044. URL: https://www.cosic.esat.kuleuven.be/apes/papers/p1030-chaum.pdf.gz (visited on 03/15/2010).

[28] H. van Rossum et al. *Privacy-Enhancing Technologies: The Path to Anonymity.* Aug. 2005. URL: http://www.ipc.on.ca/images/Resources/anoni-v2.pdf (visited on 04/01/2010).

[29] C. Ferraro. *Cost savings, security through identity management.* SearchSecurity.com. Sept. 2003. URL: http://searchsecurity.techtarget.com/news/interview/0,289202,sid14_gci928766,00.html (visited on 03/24/2010).

[30] G. Roussos and U. Patel. "Mobile Identity Management." In: *Proceeding of Mobile Business* (2002). URL: http://portal.acm.org/citation.cfm?id=1278138 (visited on 03/13/2010).

[31] S. D. Warren and L. D. Brandeis. "The Right to Privacy." In: *Harward Law Review* 4.5 (1890), pp. 193–220. URL: http://www.law.louisville.edu/library/collections/brandeis/node/225 (visited on 04/12/2010).

[32] A. F. Westin. "Privacy and freedom." In: *Business Horizons* 10.4 (1967), pp. 106–106. URL: http://ideas.repec.org/a/eee/bushor/v10y1967i4p106-106b.html (visited on 02/21/2010).

[33] American Institute of Certified Public Accountants, Inc. *2009 Top Technology Initiatives and Honorable Mentions.* 2009. URL: http://infotech.aicpa.org/Resources/Top+Technology+Initiatives/2009+Top+Technology+Initiatives+and+Honorable+Mentions.htm (visited on 04/14/2010).

[34] C. A. Ardagna et al. "Privacy Preservation over Untrusted Mobile Networks." In: *Privacy in Location-Based Applications: Research Issues and Emerging Trends.* Berlin, Heidelberg: Springer-Verlag, 2009, pp. 84–105. ISBN: 978-3-642-03510-4. DOI: http://dx.doi.org/10.1007/978-3-642-03511-1_4.

[35] Gartner, Inc. *Gartner Says Number of Identity Theft Victims Has Increased More Than 50 Percent Since 2003.* 2007. URL: http://www.gartner.com/it/page.jsp?id=501912 (visited on 04/19/2010).

[36] Grid-Tools. *Data Protection.* 2009. URL: http://www.grid-tools.com/sensitive-data-protection.php (visited on 04/19/2010).

[37] A. Goerlach, A. Heinemann, and W. Terpstra. "Survey on location privacy in pervasive computing." In: *Privacy, Security and Trust within the Context of Pervasive Computing* (2004), pp. 23–34.

[38] S. Lederer et al. "Personal privacy through understanding and action: five pitfalls for designers." In: *Personal and Ubiquitous Computing* 8.6 (2004), pp. 440–454.

[39] L. Palen and P. Dourish. "Unpacking privacy for a networked world." In: *Proceedings of the SIGCHI conference on Human factors in computing systems.* ACM. 2003, p. 136.

[40] G. Hornung and C. Schnabel. "Data protection in Germany I: The population census decision and the right to informational self-determination." In: *Computer Law & Security Report* 25.1 (2009), pp. 84–88.

[41] German Federal Constitutional Court. *BverfGE 65,1 - Volkszaehlungsurteil*. German. Dec. 1983. URL: http://www.foebud.org/video/volkszaehlungsurteil?set_language=en (visited on 04/15/2010).

[42] Federal Trade Commission. *Fair Information Practice Principles*. 2007. URL: http://www.ftc.gov/reports/privacy3/fairinfo.shtm (visited on 02/21/2010).

[43] P. Dixon. *A Brief Introduction to Fair Information Practices*. 2007. URL: http://www.worldprivacyforum.org/fairinformationpractices.html (visited on 04/16/2010).

[44] M. Hansen et al. "Privacy-enhancing identity management." In: *Information Security Technical Report* 9.1 (2004), pp. 35 –44. ISSN: 1363-4127. DOI: 10.1016/S1363-4127(04)00014-7. URL: http://www.sciencedirect.com/science/article/B6VJC-4BXN4BK-5/2/d083b1bbac41cc691d002ff66a52414e.

[45] K. Hyppönen. "Open Mobile Identity - Secure Identity Management and Mobile Payments Using Hand-Held Devices." MA thesis. University of Kuopio, 2009.

[46] J. Borking and C. Raab. "Laws, PETs and other technologies for privacy protection." In: *Journal of Information, Law and Technology* 1 (2001).

[47] M. Hansen. "Privacy-Enhancing Technologies." In: *Alexander Rossnagel (Ed.): Handbuch Datenschutzrecht* (2003), pp. 291–324.

[48] M Hansen et al. *Identity Management Systems (IMS): Identification and Comparison Study*. Tech. rep. 19960-2002-10. Unabhaengiges Landeszentrum fuer Datenschutz Schleswig-Holstein, 2003. URL: https://www.datenschutzzentrum.de/projekte/idmanage/study.htm (visited on 04/13/2010).

[49] P3P Toolbox. *What is P3P and How Does it Work*. URL: http://www.p3ptoolbox.org/guide/section2.shtml (visited on 04/14/2010).

[50] World Wide Web Consortium (W3C). *P3P 1.0: A New Standard in Online Privacy*. W3C Platform for Privacy Preferences Initiative. URL: http://www.w3.org/2000/07/p3p-brochure (visited on 04/14/2010).

[51] G. M. Køien and V. A. Oleshchuk. "Guest Editorial." In: *Privacy in Telecommunications* Volume 103 No. 2 (2007), p. 1.

[52] E. Clark. *Android market share grows at an amazing pace, Apple's hardly at all; no one notices*. 2010. URL: http://androinica.com/2010/03/20/android-market-share-grows-at-an-amazing-pace-apples-hardly-at-all-no-one-notices/ (visited on 05/25/2010).

[53] C. Ziegler. *Android's American market share soars, WinMo pays the price*. 2010. URL: http://www.engadget.com/2010/03/11/androids-american-market-share-soars-winmo-pays-the-price/ (visited on 05/25/2010).

[54] L. F. Cranor. *Web Privacy with P3P*. O'Reilly Media, Inc., 2002. ISBN: 0596003714. URL: http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20\&path=ASIN/0596003714 (visited on 04/17/2010).

[55] M. Presler-Marshal. *The Platform for Privacy Preferences 1.0 Deployment Guide.* W3C Working Group. 2002. URL: http://www.w3.org/TR/p3pdeployment (visited on 06/03/2010).

[56] J. Behrmann. *DynDNS fuer M2M mit GPRS/EDGE/UMTS?* 2009. URL: http://m2m-blog.de/2009/03/10/dyndns-fur-m2m-anwendungen-mit-gprsedgeumts/ (visited on 05/28/2010).

[57] R. Leenes, J. Schallaboeck, and M. Hansen. *Prime white paper V3.* 2008. URL: https://www.prime-project.eu/prime_products/whitepaper/PRIME-Whitepaper-V3.pdf (visited on 07/22/2010).

[58] L. Cranor et al. *The Platform for Privacy Preferences 1.1 (P3P1.1) Specification.* W3C Working Group. 2006. URL: http://www.w3.org/TR/P3P11/ (visited on 05/07/2010).

[59] J. Freed. *Getting Your Web Site P3P Compliant.* Internet Education Foundation. June 2009. URL: http://www.p3ptoolbox.org/tools/papers/IEFGettingyoursitecompliant.ppt (visited on 05/07/2010).

[60] T. Berners-Lee, R. Fielding, and L. Masinter. *RFC3986 - Uniform Resource Identifiers: Generic Syntax and Semantics.* IETF. 1998. URL: http://www.ietf.org/rfc/rfc3986.txt (visited on 05/18/2010).

[61] P3P Toolbox. *Preparing for the P3P Implementation.* URL: http://www.p3ptoolbox.org/guide/section3.shtml (visited on 06/04/2010).

# APPENDIX A

# Appendices

This appendix includes visualizations of the finalized procedural framework method and the process sequence within the anonymous network. It then presents additional fundamentals related to the P3P technology. This is followed by a definition of typical privacy level evaluations in the corresponding analyzing process. After that, identification rules are stated that allow to determine privacy risks and issues before identity attributes are disclosed. Finally, a brief primer into the working area within IP addresses is provided.

# A.1 Finalized Procedural Framework Method

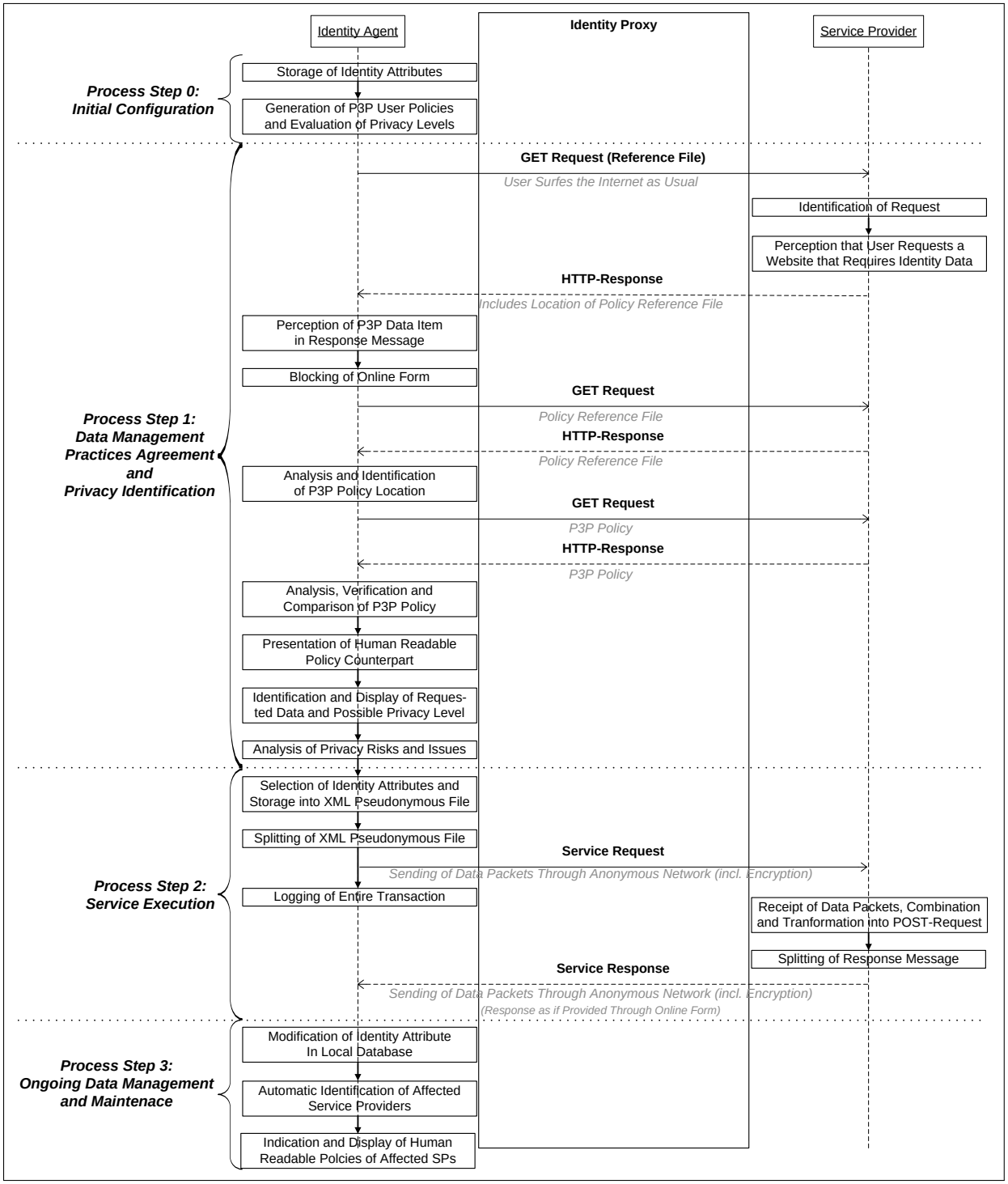



Figure A.1: Finalized procedural framework method (high resolution)

## A.2 Process Sequence within the Anonymous Network

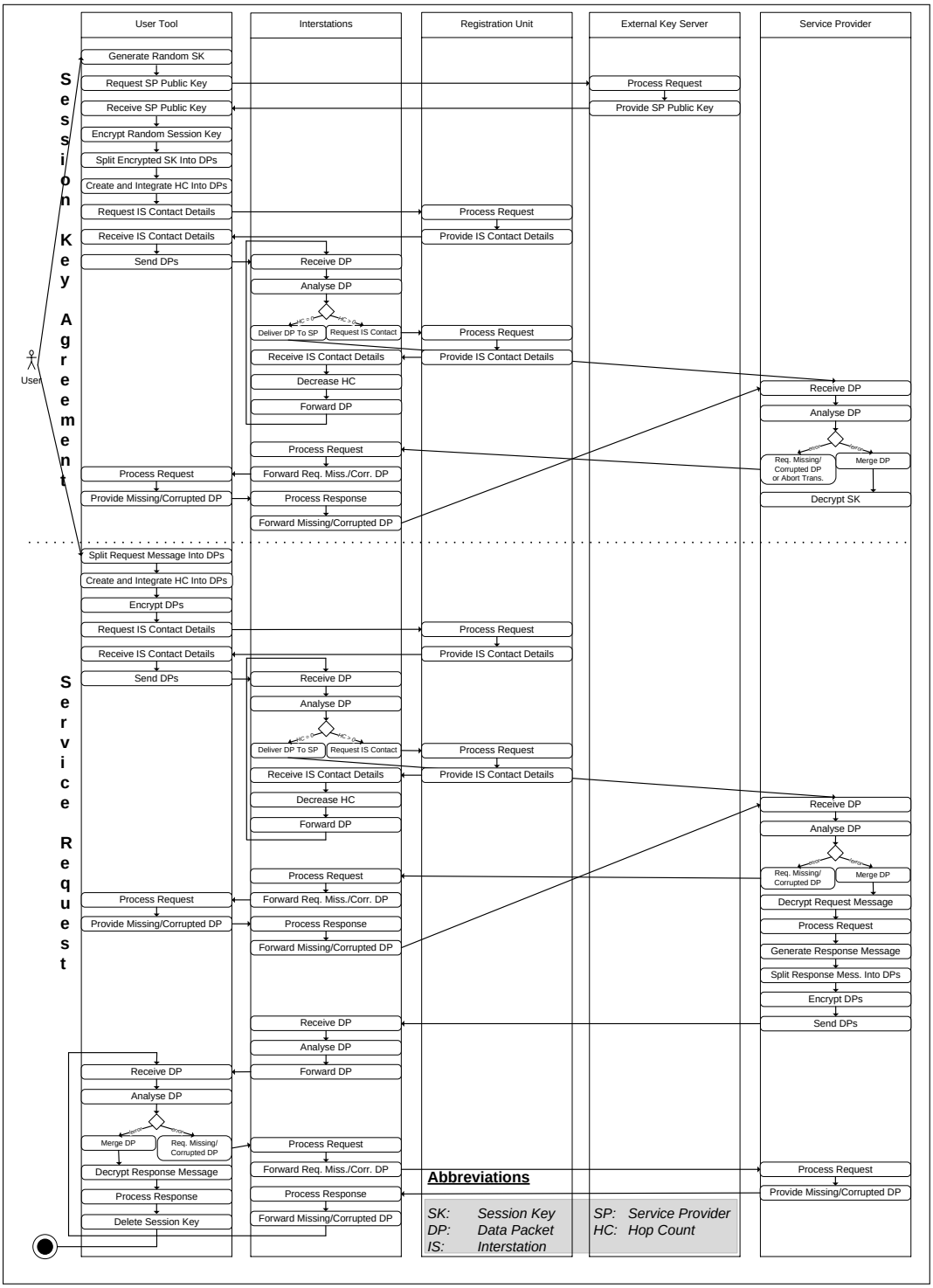



Figure A.2: Process sequence within the anonymous network

## A.3 P3P Technology

This appendix describes the P3P technology on that every privacy related framework process is based. It presents the applied vocabulary including a project dependent extension and the applied data schema. It then shows a typical P3P policy to be used on SP sides and defines a brief step by step guide for related policy generation. The user interface to generate policies on mobile phones (UI2) then concludes this appendix.

### A.3.1 Applied Vocabulary

This subsection shows the P3P vocabulary that enables generation of data management policies. Thereby, it only extracts project relevant aspects, rather than presenting the entire vocabulary. It also extends the *<ACCESS>* as described in Subsection 3.3.1. For more information than presented here it is referred to the two used sources [58] and [59].

**<ENTITY>**

- Precisely describes the organization that is collecting data and to that the policy belongs to.
- Needs to include the company name and one or more information belonging to the *business* data element[1].

**<ACCESS>**

- Defines whether or not the site provides users access to collected data.
- SPs are required to define this attribute.
- No need to specify the method of access (to find out the ways, users are required to contact the SP, e.g. by locating the human readable privacy policy). "P3P does not include a mechanism to automate data access or update" [59].
- Parameter determines the type of accessible data; possible parameters (only one selectable):
  - **<all/>** - Access to all identity attributes.
  - **<ident-contact/>** - Access to online and physical contact information (e.g. postal address).
  - **<none/>** - No access at all.

**<DISPUTES>**

- Lists methods to resolve disputes related to privacy practice violations.
- "Should" be included in a P3P policy.
- Mandatory attributes:
  - **Resolution-Type** (only one selectable):
    - **[service]** - Company's customer service is available (contact information required).
    - **[independent]** - Independent organization is available (contact information required). Can also be used to state achieved privacy seals and certification programs.
    - **[court]** - Court authority is available (needs to be specified).
    - **[law]** - Laws or regulations are available (need to be specified).

**<PURPOSE>**

- Specifies the purpose of using collected data.
- Required to contain one or more of the following purposes:

[1] Whenever this subsection mentions data elements it is refereed to the P3P data schema of Appendix A.3.2.

  - <**current**/> - Completion and support of actual transaction.
  - <**admin**/> - Technical support of web site and system administration.
  - <**develop**/> - Enhancement, evaluation or review of site, service, product, or market.
  - <**pseudo-analysis**/> - Creation of user or computer record, without tying identified data (name, address, phone number, or email address) to the record. Record will be used for determination of habits, interests or other characteristics for research, analysis or reporting approaches but not used to identify individuals.
  - <**individual-analysis**/> - Determination of habits, interests, or other characteristics of individuals. Combination with identified data for research, analysis or reporting approaches.
  - <**contact**/> - Contacting of users for marketing or service approaches (e.g. notification of web site updates). Marketing via customized web content or banner advertisements is not included.
  - <**telemarketing**/> - Contacting of users for marketing via telephone.
  - <**other-purpose**> - Purpose that has not been covered by the other definitions. A human-readable explanation is required.
- Each type of purpose (with exception of *current*) can have the following optional attribute:
  - **always** - Purpose always required, users cannot opt-in or opt-out. Default selection when no attribute is chosen.
  - **opt-in** - Purpose only affective if user actively requests it.
  - **opt-out** - Purpose affective unless user actively declines it.

<**RECIPIENT**>

- Determines the recipients of collected data.
- SPs are required to define this attribute.
- Required to contain one or more of the following recipients (including optional description):
  - <**ours**/> - Company itself and agents or entities for whom the company is acting (e.g. SP and its printing bureau).
  - <**delivery**/> - Delivery services (this element "should not" be used if delivery services agree to use collected data only for completion of SP delivery).
  - <**same**/> - Legal entities following the same privacy practices as collecting company.
  - <**other-recipient**/> - Legal entities that are acting under different privacy practices than collecting company.
  - <**unrelated**/> - Legal entities whose privacy practices are not known by collecting company.
  - <**public**/> - Public fora (e.g. bulletin boards, public directories, or commercial directories).
- With exception of *<ours/>*, required to indicate availability of opt-in or opt-out choices.

<**RETENTION**>

- Defines the duration of storing collected data in general terms, no specific statements.
- Detailed information about the stated retention practice needs to be included into the human readable policy.
- Required to contain one of the following values:
  - <**no-retention**/> Information is not retained for more than a brief period of time necessary to make use of it during the course of a single online interaction. Information MUST be destroyed following this interaction and MUST NOT be logged, archived, or otherwise stored. T
  - <**stated-purpose**/> - Information discarded at the earliest point of time possible.
  - <**legal-requirement**/> - Information stored according to laws and legal requirements.
  - <**indefinitely**/> - Information stored for an indeterminate period of time (e.g. public fora).

<**DATA-GROUP**>

- Lists the specific data collected by the company under the stated privacy practices.

- Grouping is carried out with the help of the P3P data schema (see Subsection A.3.2).
- Sites can describe the data they collect using either specific data elements, or simply by categories of data.
- Not required is statement whether submission of data element is optional to access resource or complete transaction:
  - **no** - Data element required.
  - **yes** - Data element optional.

**<STATEMENT>**

- Container to group a <PURPOSE>, <RECIPIENT>, <RETENTION> and <DATA-GROUP> element
- Same privacy practices apply to all included data.

**<CATEGORIES>**

- Helps to categories data elements, so that expression of more generalized privacy preferences is possible.
- See also Subsection A.3.2 for possible categorizations:
  - <**physical**> - Physical Contact Information.
  - <**online**> - Online Contact Information.
  - <**uniqueid**>: Unique Identifiers, including Login Information.
  - <**purchase**>: Purchase Information.
  - <**financial**>: Financial Information.
  - <**demographic**>: Demographic and Socioeconomic Data.
  - <**locational**>: Location and Time-Based Data (project individual extension).
  - <**other-category**>: Category not captured by the others.

### A.3.2 Applied Data Schema

After the P3P vocabulary is clarified it is necessary to show the related P3P data schema. As the previous subsection this one is also based on [58]. The data schema helps to describe the information that SPs want to collect. Thereby, the following Table A.1 only presents data elements that are relevant for this project.

Table A.1: Applied data schema

| Type | Allowed Categories | Allowed Descendants | Short Description | Notes |
|---|---|---|---|---|
| user | Physical Contact Information, Demographic and Socioeconomic Data, Unique Identifiers, Online Contact Information | name, bdate, login, gender, home-info, business-info | General information about the user | |

Table A.1: Applied data schema (cont.)

| Type | Allowed Categories | Allowed Descendants | Short Description | Notes |
|---|---|---|---|---|
| third-party | Physical Contact Information, Demographic and Socioeconomic Data, Unique Identifiers, Online Contact Information | name, business-info | | Related third parties. Useful whenever third party information needs to be exchanged (e.g. ordering an item that should be sent to another person, or providing information about one's spouse or business partner). |
| business | | orgname, department, contact-info | | Subset of *user* data relevant for describing legal entities. In P3Pv.1.1 primarily used for describing the policy entity. |
| orgname | Physical Contact Information, Demographic and Socioeconomic Data | | Organization Name | |
| name | Physical Contact Information, Demographic and Socioeconomic Data | prefix, given, family, nickname | User's Name | |
| bdate | Demographic and Socioeconomic Data | ymd.year, ymd.month, ymd.day | User's Birth Date | |
| login | Unique Identifiers | id, password | User's Login Information | IDs and passwords for computer systems and Web sites which require authentication. |
| gender | Demographic and Socioeconomic Data | | User's Gender (Male or Female) | |
| department | Demographic and Socioeconomic Data | | User's Department or Division in the Organization | |
| home-info | Physical Contact Information, Online Contact Information, Demographic and Socioeconomic Data | postal, telecom, online | User's Home Contact Information | |
| contact-info | Physical Contact Information, Online Contact Information, Demographic and Socioeconomic Data | postal, telecom, online | Contact Information of the Organization | |

Table A.1: Applied data schema (cont.)

| Type | Allowed Categories | Allowed Descendants | Short Description | Notes |
|---|---|---|---|---|
| business-info | Physical Contact Information, Online Contact Information, Demographic and Socioeconomic Data | postal, telecom, online, department | User's Business Contact Information | |
| ymd.-year | *variable-category* | | Year | |
| ymd.-month | *variable-category* | | Month | |
| ymd.day | *variable-category* | | Day | |
| hms.-hour | *variable-category* | | Hour | |
| hms.-minute | *variable-category* | | Minute | |
| hms.-second | *variable-category* | | Second | |
| prefix | Demographic and Socioeconomic Data | | Name Prefix | |
| given | Physical Contact Information | | Given Name (First Name) | |
| family | Physical Contact Information | | Family Name (Last Name) | |
| nick-name | Demographic and Socioeconomic Data | | Nickname | |
| id | Unique Identifiers | | Login ID | ID portion of login information for computer systems. Often, user IDs are made public, while passwords are kept secret. |
| pass-word | Unique Identifiers | | Login Password | Password portion of login information for computer systems. Secret data value, usually character string, that is used for authenticating a user. Passwords are typically kept secret and generally considered to be sensitive. |
| postal | Physical Contact Information, Demographic and Socioeconomic Data | name, street, city, stateprov, postalcode, country | Postal Address Information | |
| telecom | Physical Contact Information | telephone, fax, mobile | Telecommunication Information | Characteristics of telephone, fax and mobile numbers. |
| online | Online Contact Information | email, uri | Online Address Information | Online information about a person or legal entity. |

Table A.1: Applied data schema (cont.)

| Type | Allowed Categories | Allowed Descendants | Short Description | Notes |
|---|---|---|---|---|
| intcode | Physical Contact Information | | International Telephone Code | |
| loccode | Physical Contact Information | | Local Telephone Area Code | |
| number | Physical Contact Information | | Telephone Number | |
| street | Physical Contact Information | | Street Address | |
| city | Demographic and Socioeconomic Data | | City | |
| state-prov | Demographic and Socioeconomic Data | | State or Province | |
| postal-code | Demographic and Socioeconomic Data | | Postal Code | |
| country | Demographic and Socioeconomic Data | | Country Name | Country name (e.g., one among countries listed in ISO3166[2]). |
| organization | Demographic and Socioeconomic Data | | Organization Name | |
| telephone | Physical Contact Information | intcode, loccode, number | Telephone Number | |
| fax | Physical Contact Information | intcode, loccode, number | Fax Number | |
| mobile | Physical Contact Information | intcode, loccode, number | Mobile Telephone Number | |
| email | Online Contact Information | | Email Address | |
| uri | Online Contact Information | | Home Page Address | URI as defined in [60]. |

[2]More information about ISO3166 given by *English country names and code elements* at `http://www.iso.org/iso/english_country_names_and_code_elements`.

### A.3.3 Typical Policy

Listing A.1 shows a typical P3P policy that can be published on SP sides.

```
<POLICIES xmlns="http://www.w3.org/2002/01/P3Pv1">  <!-- XML Namespace -->
<POLICY name="youbuy_order"
    opturi="http://www.YouBuy.com/P3P/optin.html" <!-- URI of instructions to
        request or decline using collected data (opt-in or opt-out) -->
    discuri="http://www.YouBuy.com/P3P/disc.html" <!-- URI of the privacy policy
        in human readable format -->
     xml:lang="en">

<!-- Entity making this policy statement -->
<ENTITY>
  <DATA-GROUP>
    <DATA ref="#business.name">YouBuy</DATA>
    <DATA ref="#business.contact-info.postal.street">Grooseveien 36</DATA>
    <DATA ref="#business.contact-info.postal.city">Grimstad</DATA>
    <DATA ref="#business.contact-info.postal.postalcode">4876</DATA>
    <DATA ref="#business.contact-info.postal.country">Norway</DATA>
    <DATA ref="#business.contact-info.online.email">contact@YouBuy.com</DATA>
    <DATA ref="#business.contact-info.telecom.telephone.intcode">47</DATA>
    <DATA ref="#business.contact-info.telecom.telephone.loccode">37</DATA>
    <DATA ref="#business.contact-info.telecom.telephone.number">123456</DATA>
  </DATA-GROUP>
 </ENTITY>

<!-- Provided user access to collected data -->
<ACCESS><ident-contact/></ACCESS>

<!-- User methods to resolve disputes related to privacy practice violations -->
<DISPUTES-GROUP>
  <DISPUTES resolution-type="service" service="http://www.YouBuy.com/contact.asp"
     short-description="Dispute Resolution by contact form">
    <LONG-DESCRIPTION>We will work to resolve any issues with this privacy policy
        .
    </LONG-DESCRIPTION>
  </DISPUTES>
</DISPUTES-GROUP>

<!-- Statement for group "Customer Details", same privacy practices apply to
    every single data included -->
<STATEMENT>
    <!-- Provides a human-readable explanation that can be presented to users by
        user tool -->
    <CONSEQUENCE> In order to deliver items accordingly customer details of our
        are
   collected. </CONSEQUENCE>
   <!-- Company's purpose of using collected data) -->
   <PURPOSE> <current/><contact/><telemarketing/> </PURPOSE>
   <!-- Recipients of collected data -->
   <RECIPIENT> <ours/> <other-recipient  required="opt-out"/> </RECIPIENT>
   <!-- Duration of storing collected data -->
   <RETENTION> <legal-requirement/> </RETENTION>
   <!-- Base data schema elements -->
   <DATA-GROUP>
     <DATA ref="#user.name.prefix"/>
     <DATA ref="#user.name.given"/>
```

```
49      <DATA ref="#user.name.family"/>
50      <DATA ref="#user.bdate.ymd.year"/>
51      <DATA ref="#user.bdate.ymd.month"/>
52      <DATA ref="#user.bdate.ymd.day"/>
53      <DATA ref="#user.gender"/>
54      <DATA ref="#user.home-info.postal.street"/>
55      <DATA ref="#user.home-info.postal.city"/>
56      <DATA ref="#user.home-info.postal.postalcode"/>
57      <DATA ref="#user.home-info.postal.country"/>
58      <DATA ref="#user.home-info.telecom.telephone.loccode"/>
59      <DATA ref="#user.home-info.telecom.telephone.number"/>
60      <DATA ref="#user.home-info.online.email"/>
61    </DATA-GROUP>
62 </STATEMENT>
63
64 </POLICY>
65 </POLICIES>
```

Listing A.1: "Typical P3P policy for publication on SP side"

### A.3.4 Step By Step Guide: Policy Generation

The following seven-steps guide eases the process of generating P3P policies, based on [61].

1. Define the entity that is collecting data.
2. Specify the location of the human readable policy.
3. Describe identity attributes to be collected with the help of the P3P data schema (Appendix A.3.2); grouping of attributes helps to create data categories.
4. Define the purpose for collection and processing.
5. Categorize third party recipients that are allowed to access collected data.
6. Clarify availability of opt-in and opt-out choices.
7. Specify dispute resolution, data retention and access.

### A.3.5 User Interface to Generate Policies

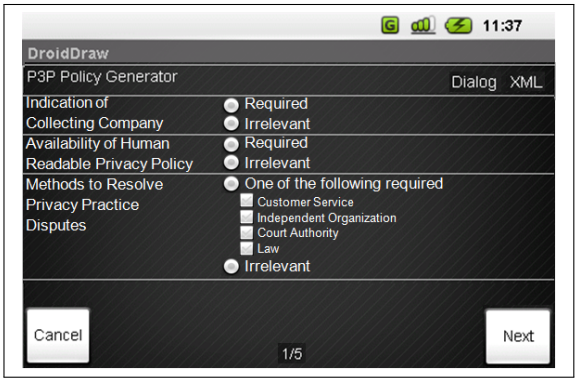



Figure A.3: User interface to generate P3P policies, Page 1

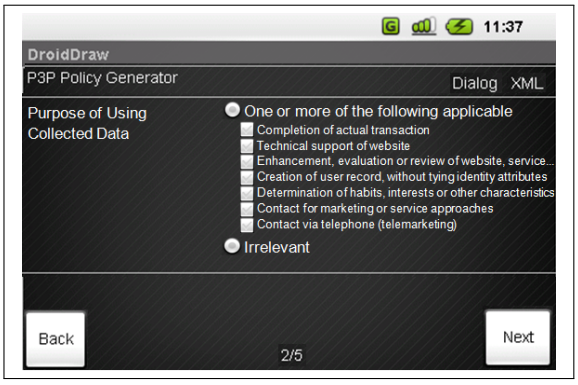



Figure A.4: User interface to generate P3P policies, Page 2

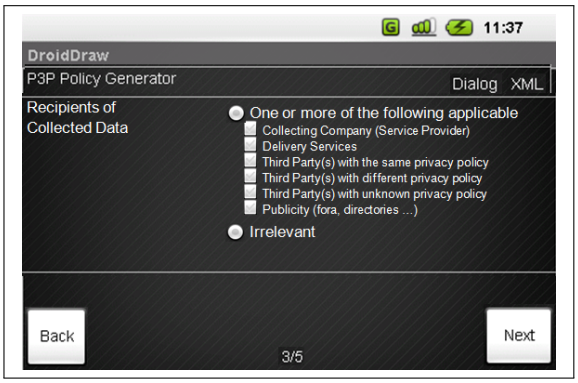



Figure A.5: User interface to generate P3P policies, Page 3

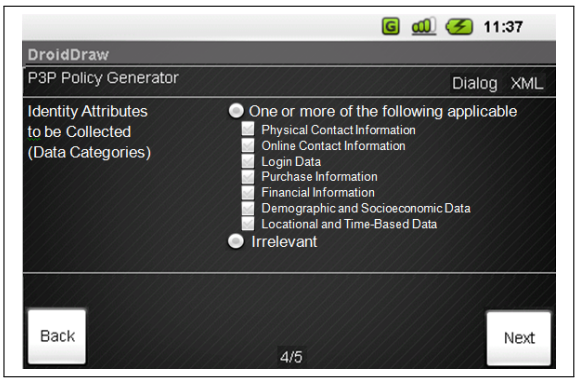



Figure A.6: User interface to generate P3P policies, Page 4

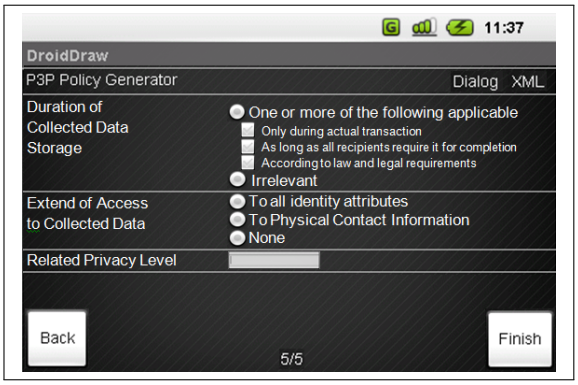



Figure A.7: User interface to generate P3P policies, Page 5

## A.4 Typical Privacy Levels and Risk Identification Rules

This appendix works closely connected to the user tool design. It starts with a presentation of typical privacy level evaluations and the related analyzing process. It then shows sample identification rules that allow to automatically determine predefined privacy risks and issues.

### A.4.1 Typical Privacy Level Evaluations

The following Table A.2 shows typical evaluations of different privacy levels. However, it is not complete and only aims to give a brief impression of related capabilities. Furthermore, differing opinions can lead to other evaluations.

Whenever a star (*) is presented, it indicates that all available attributes are applicable. The only limitation is that it is not allowed to leave the field blank. The two horizontal lines (//) then define that one or more of the listed attributes can be specified in order to reach the depicted privacy level.

Table A.2: Typical privacy level evaluations

| **Human Policy** | **Entity** | **Access** | **Disputes** | **Purpose** | **Recipient** | **Retention** | **Categories** | **Privacy Level** |
|---|---|---|---|---|---|---|---|---|
| * | * | * | * | * | public | * | * | **0** |
| * | not specified | * | * | * | * | * | * | **0** |
| n.a. | specified | none | * | current \|\| admin \|\| develop \|\| pseudo-analysis \|\| individual-analysis \|\| contact \|\| telemarketing | unrelated | indefinitely | physical \|\| online \|\| purchase \|\| financial \|\| demographic | **1** |
| ✓ | specified | none | | current \|\| admin \|\| develop \|\| pseudo-analysis \|\| individual-analysis \|\| contact \|\| telemarketing | other-recipient | indefinitely | physical \|\| online \|\| purchase \|\| financial \|\| demographic | **2** |
| ✓ | specified | other-ident | * | current \|\| admin \|\| develop \|\| pseudo-analysis \|\| individual-analysis \|\| contact \|\| telemarketing | other-recipient | legal-requirements | physical \|\| online \|\| purchase \|\| financial \|\| demographic | **3** |
| ✓ | specified | other-ident | * | current \|\| admin \|\| develop \|\| pseudo-analysis \|\| individual-analysis | other-recipient | legal-requirements | physical \|\| online \|\| purchase \|\| financial \|\| demographic | **4** |
| ✓ | specified | other-ident | * | current \|\| admin \|\| develop | ours \|\| same \|\| delivery | legal-requirements | physical \|\| online \|\| purchase \|\| financial \|\| demographic | **5** |
| ✓ | specified | ident-contact | * | current \|\| admin | ours \|\| same \|\| delivery | legal-requirements | physical \|\| online \|\| purchase \|\| financial \|\| demographic | **6** |
| ✓ | specified | ident-contact | * | current | ours | legal-requirements | physical \|\| online \|\| purchase | **7** |
| ✓ | specified | contact-and-other | * | current | ours | stated-purpose | physical \|\| online | **8** |
| ✓ | specified | all | * | current | ours | stated-purpose \|\| no-retention | physical \|\| online | **9** |
| ✓ | specified | all | * | current | ours | no-retention | physical | **10** |

### A.4.2 Privacy Level Identification

This paragraph is an addition to the discussion of UTM1 in Subsection 3.3.4. Thus it presents the comparison of Carol's (Listing 3.7) and YouBuy's privacy policy (Listing A.1) in Table A.3. Thereby, it is identified that the included requirements and specifications for a privacy level of 7 match to 58%.

Table A.3: Comparison of Carol's and YouBuy's policies for a privacy level of 7

| **Carol's Specification** | **YouBuy's Specification** | **Match** | **Mism.** |
|---|---|---|---|
| discuri="*" | discuri="http://www.YouBuy.com/..." | **x** | |
| <ENTITY> <DATA-GROUP> | <ENTITY> <DATA-GROUP> | | |
| ..."#business.name"... | ...#business.name">YouBuy... | **x** | |
| ...#business.contact-info.postal.*"... | ...#business.contact.info.postal.street">Groos... | **x** | |
| ...#business.contact-info.telecom.telephone.*... | ...telecom.telephone.intcode">47... | **x** | |
| </DATA-GROUP> </ENTITY> | </DATA-GROUP> </ENTITY> | | |
| <ACCESS><ident-contact/></ACCESS> | <ACCESS><ident-contact/></ACCESS> | **x** | |
| <PURPOSE> <current/> </PURPOSE> | ...<current/><contact/><telemarketing/>... | | **x** |
| <STATEMENT> <DATA-GROUP> | <STATEMENT> <DATA-GROUP> | | **xx** |
| ..."#user.name.*... | ...#user.name.given"... | **x** | |
| ..."#user.home-info.postal.*"... | ..."#user.home-info.postal.street"... | **x** | |
| | ..."#user.bdate.ymd.month"... | | **x** |
| | ..."#user.gender"... | | **x** |
| </DATA-GROUP> </STATEMENT> | </DATA-GROUP> </STATEMENT> | | |
| | **Total** | **7** | **5** |
| **58%** *(percentage match factor)* | | | |

### A.4.3 Typical Identification Rules for Privacy Risks and Issues

Table A.4 works together with Subsection 3.3.4. It provides typical identification rules that allow to analyze privacy risks and issues. Their aim is to cover the Privacy Design Requirements that are shown in Table 3.1. Due to their alignment to the P3P vocabulary simplified coding and integration into the user tool is supported.

## A.5 Internet Protocol Addresses

This appendix works as a primer into the working area within IP addresses. It is not uncommon to compare IP addresses to telephone numbers. When a person wants to talk to another one he can reach him by a unique number. The same applies to computers - every system that is connected to a network owns an individual IP address that identifies and differentiates it from others.

Table A.4: Typical identification rules for privacy risks and issues

| Situation | Privacy Risk or Issue |
|---|---|
| Same SP requests differing data categories than collected previously. | User data concatenation. |
| Same SP requests same category but differing identity attributes than collected previously. | User data concatenation. |
| Same SP, same transactional purpose but differing identity attributes than collected previously. | Attribute change. User data concatenation. |
| SP changes specification of recipients. | Recipient change. |
| SP changes specification of retention. | Retention change. |
| SP changes specification of access. | Access change. |
| SP changes specification of disputes. | Disputes change. |
| SP changes specification of purpose. | Purpose change. |
| SP requests locational information. | Locational disclosure. |
| Same SP requests differing identity attributes belonging to location data than collected previously. | Position tracking. |
| SP declines access to collected data. | Denial of reviews and modifications on SP sides. |

### A.5.1 Available Address Types

Talking about IP addresses there are different types available that specify various applicabilities. On the first hand *private* addresses are only accessible from computers within the same network. This type is often used for private networks that do not require accessibility from the outside. On the other hand, to also enable availability from other networks *public* addresses are used. Their visibility is not limited to one particular local network.

Moreover, public IP addresses are further distinguished by their lifetimes. So-called *dynamic* addresses are renewed with every dial-up, whereas *static* ones never change. This means, that computers with static IP addresses can be contacted with one and the same number at any time.

Computers that provide services over the internet need unique addresses with broad applicability. Thus, static public ones are best suitable. However, as a reason of limited numbers of available IP addresses this is not always realizable[3]. Therefore, these addresses mostly require to be bought from the Network Service Provider.

[3]With launch of Internet Protocol Version 6 (IPv6) addresses a much larger address pool will be available.

### A.5.2 Using DNS Names

To make handling of IP addresses easier and to unburden users from the need to remember lots of different addresses, DNS was introduced. A so-called *DNS name* is an identifier that maps IP addresses to host names. Thus, webpage addresses (e.g. `http://www.google.com`) are used because they are much easier to remember than a bunch of numbers. In the background web browsers automatically map the provided names to corresponding IP addresses. The same applies to computer networks in that every system can be uniquely identified by it's host name (e.g. PC1 and PC2). Responsible for resolving IP addresses into DNS names (and the other way round) are DNS servers.

DNS thus seems in first instance appropriate for computers that provide services and at the same time do not own static public IP addresses. However, the concept fails when the public address changes, and that is the case without the static characteristic. This is the reason, because DNS servers are not automatically notified of address changes. Therefore, there is a risk that they provide systems with expired IP addresses.

To overcome the this issue online services like *DynDNS*[4] were designed. Such service are able to automatically register IP address changes and to update their databases immediately. Using DynDNS thus enables users to contact participating computers with one and the same DNS name over time, regardless whether the public (dynamic) IP address has changed or not. Thus, as a conclusion, such a service is a good alternative to buying static public IP addresses.

[4]DynDNS is just one of many identical services. It enjoys great popularity and is also free to use. See the manufactures website at `https://www.dyndns.com/`.