Trusted Secure Service Design

Master of Science Thesis

In Security and Mobile Computing

From the NordSecMob Erasmus Mundus Programme

by

Thomas C. M. Vilarinho

Trondheim, June 2009

External Supervisor: Josef Noll

Industry Supervisor: Kjetil Haslum

KTH Supervisor: Peter Sjödin

NTNU Supervisor: Øivind Kure

Abstract

The SIM cards are going through several new enhancements both in the underlying hardware and its capabilities. They are becoming secure wireless networked devices containing embedded sensors. This thesis assess how this new SIM capabilities together with its pervasiveness and security can support the development and design of trust-based applications. It reviews the new trust possibilities based on the identity factor, connectivity and context-awareness sensors on the SIM. Moreover, we present a specific use-case around a seamless trust builder for social networks, which makes use of sensed inputs towards building hard contextual evidences to trust relations. We conclude with the description of the challenges of building this evidence based trust-builder and the necessary steps to going from the prototype we developed to a real application which may accurately describe trust relations.

Keywords: SIM cards, trust, networked embedded systems, pervasive computing, Sun SPOT, identity management**,** social networks, context-awareness

Sammanfattning

SIM-korten utvecklas och genomgår förbättringar i den underliggande hårdvaran och dess egenskaper. Korten håller på att bli säkra trådlösa enheter som innehåller inbyggda sensorer. Denna rapport handlar om hur dessa nya egenskaper, inklusive identitietsaspekter och omgivningsberoende sensorer på SIM-korten, i kombination med kortens genomslagskraft och säkerhetsegenskaper, kan ge stöd för utveckling och design av tillämpningar som baseras på förtroenderelationer. Rapporten beskriver också ett specifikt användarfall kring en integrerad enhet för etablerande av förtroenderelationer i sociala nätverk, vilken använder indata från sensorer för att bygga upp starka kontextuella belägg för förtroenderelationer. Avslutningsvis beskrivs de speciella utmaningar som det innebär att konstruera en sådan enhet, samt de steg som krävs för att gå från den prototyp som utvecklats till en verklig tillämpning som kan beskriva förtroenderelationer på ett korrekt sätt.

ACKNOWLEDGMENTS

With this thesis I conclude the Erasmus Mundus Master Programme NordSecMob. The thesis was done physically in the Telenor R&I office in Trondheim, and it counted with the supervision of Royal Institute of Technology (KTH), Norwegian University of Science and Technology (NTNU), Telenor R&I and University Graduate Center (UNIK). The result of this semester work is presented in this report and in a paper submitted to the NordSec 2009 Conference.

The thesis was supervised by professor Josef Noll from Unik and Kjetil Haslum from Telenor R&I, in conjunct with the academic guidance of professor Øivind Kure from NTNU and Peter Sjödin from KTH. I would like to thank them all for the support and the time spent on the thesis; and especially Josef for the moments of guidance and Kjetil for the continuous day-to-day follow-up.

I would like to thank others that helped with discussions, information and by providing feedback on the thesis: Steinar Brede, Rehab Alnemr, Tor-Hjalmar Johannessen, Brynjar Viken, Narada Dilp Warakagoda, Fritjof Boger Engelhardtsen, Karin Sallhammar, Jan Marius Sund and Jan Partanen to mention some. Besides that I would like to express my gratitude to the coordinators of the NordSecMob program, in special to Mona Nardaune, May-Britt Eklund-Larsson and Eija Kujanpää, which assisted me in several situations during the two years of the Master.

At last, I would like to thank my friends, my family and my girlfriend for the support and encouragement during those two years and especially while writing the thesis.

TABLE OF CONTENTS

FIGURE LIST

TABLE LIST

ACRONYMS

APDU: Application Protocol Data Unit

BIP: Bearer Independent Protocol

CA: Certification Authority

CellId: Cell Identification

EEM: Ethernet Emulation Mode

E-Id: electronic ID

ETSI: European Telecommunications Standards Institute

FINEID: Finnish Electronic Identity

GPS: Global Positioning System

GSM: Global System for Mobile Communications

GSMA: GSM Association

HSM: Hardware Security Module

IdM: Identity Management

IdP: Identity Provider

IMSI: International Mobile Station Identifier

J2ME: Java 2 Platform, Micro Edition

M2M: Machine to Machine

ME: Mobile Equipment

MMC: Multimedia Card

MOC: Match-on-Card

NFC: Near Field Communication

OSN: Online Social Network

OTA: Over-the-Air

OTP: One-Time-Password

PKI: Public Key Infrastructure

RAM: Remote Applet Management

REST: Representational State Transfer

RMI: Remote Method Invocation

RSSI: Received Signal Strength Indication

SAML: Security Assertion Markup Language

SAT: SIM Application Toolkit

SATSA: Security and Trust Services API

SCWS: Smart Card Web Services

SD: Security Domain

SIM: Subscriber Identity Module

SMS-PP: Short Message Service - Point to Point

SOA: Service-Oriented Architecture

SSO: Single Sign On

STK: SIM Toolkit (the same as SAT)

Sun SPOT: Sun Small Programmable Object Technology

SWP: Single Wire Protocol

UICC: Universal Integrated Circuit Card

WIB: Wireless Internet Browser

WSDL: Web Services Description Language

1. INTRODUCTION

1.1. Thesis Definition

In short words, this thesis aims to assess the trust building based on Future SIM features, in special its identity capabilities and context awareness power. Then, we choose one of the identified trust scenarios for the Future-SIM and we implement a prototype application, in order to develop a proof-of-concept on the trust enhancement of the SIM.

The new contributions to the study area can be summarized in:

- Assessment of future and current trends for the SIM cards.
- Review of SIM-based identity management and relationship with industry standards.
- Assess trust modeling and identify how the future SIM can be used in it.
- Implementation of a proof-of-concept application building trust with the Future SIM.

The main focus of the thesis is on the theoretical part, where I analyzed capabilities of a future SIM and current trends in industry. The trust inference application implemented should be seen as a prototypical implementation, as it was reserved just around three weeks of the whole thesis for it.

1.2. Motivation

It is unanimous that the mobile phone is the most popular personal pervasive device so far. The strong presence of mobile phones, together with the development of new interfaces and sensors, has pushed several applications to be developed on this platform. However, the mobile phone itself is not considered a secure platform as it seldom has memory access protection and physical tampering sensors.

On the other hand all mobile devices represented by the Global System for Mobile communication (GSM), which represent more than 80% of the mobiles[1], have a security element represented as the Subscriber Identity Module (SIM) card. The SIM as a smart card corresponds to a well trusted and tamper-proof device. Smart cards are trusted enough to play

[1] http://www.gsmworld.com/newsroom/market-data/market_data_summary.htm

key roles in highly secure business cases such as banking, key management, identification and authentication.

Besides the access to all features from the mobile or from any other device to which the SIM card is connected, new physical and logical interfaces are becoming available to the SIM. Standard and non-standard wired and wireless interfaces and sensors are being integrated to the SIM. In addition to that, we are starting to see more and more cases of multi-application cards.

These new capabilities being developed around the SIM, allied with its pervasiveness and security, enable it to act as a trust component for federated identities, or as a secure platform for the Internet-of-things, and to provide seamless contextual evidences or a mix of those roles. Thus, this thesis aims to identify new range of usages of the SIM.

1.3. Report Organization

This report is divided in 11 chapters including this introduction. In the introduction we define the topic that we worked during this semester and what are the goals and contributions of this Master Thesis.

The second chapter presents the methodology followed in order to accomplish the established goals. It describes the plan and internal deadlines for the thesis.

The Future SIM assessment is divided in the “Smart Card” chapter, the “SIM Card” chapter and the “Sensors and Context” chapter. The first one gives an introduction to smart cards and it goes into details on the security features of the smart cards. The “SIM Card” chapter enumerates the differences from the SIM Card to regular smart cards, focusing on its interfaces towards the mobile phone and other platforms. It also reviews the new trends around the SIM based on publications from related institutions and researchers. At last, the “Sensors and Context” presents context information and it explores the methods that the Future SIM can use to become a context-aware device.

In the chapter 6 we present the concept of identities and we explore some identity and IdM implementations varying from the governmental E-Id smart cards, to industry frameworks and the GSMA IdM framework for the SIM card.

The Trust chapter presents the concept of trust and both the policy-based and reputation-based trust modes. Besides that, it discusses the trust relation in online social networks, as we decided to tackle trust in that area.

Chapter 8 summarizes the scenarios we identified as the ones that could greatly benefit on the Future-SIM capabilities to enhance their trust. Moreover, there we present the seamless trust builder case that we decided to implement.

The chapters 9 and 10 present the application implemented and the experiments performed. In the implementation chapter we focus on the description of the application, its logic and some simplifications we had to do. In the other hand, the experiments chapter describes the emulated scenario; it describes the strong and weak points of our application as it goes through the experimentation steps.

We sum it up our finding on the conclusion chapter and we present some future work that could be done based on our achievements.

2. Methodology

In this chapter we present the methodology used in the development of the thesis. We describe how the work was organized and planned in order to achieve its results.

2.1. Workflow organization

This thesis has been carried out physically at the Telenor R&D department in Trondheim, where Kjetil Haslum, Steinar Brede and some other Telenor SIM card experts were consulted for technical discussions and guidance on the thesis. Once the external supervisor and main guide in the thesis, Josef Noll, has his office in Kjeller, the communication was done through weekly reports and eventually through phone calls or presence meetings.

A wiki from Unik[2] has been used in order that all parties could be aware of the thesis progression and achievements. During each study phase, I have documented in the wiki a structured summary that could afterwards be easily ported to the final thesis report. The supervisors could then keep an eye on the thesis achievements and also in the bibliography used in order to assure a good quality of the work and that there are no plan deviations.

And, at the end of the week, I have published a small weekly report and the plans for the following week. This approach worked really well and helped me to accomplish our goals in time. The thesis wiki can be found at http://wiki.unik.no/index.php/Thesis/TrustedService.

2.2. Thesis project plan

We divided the thesis into four main parts:

- Future SIM Study: Where we have done an assessment of the Future SIM based on the latest experiments and research on going on the SIM cards. We consulted market news, drafts and proposals of related organizations such as ETSI and GSMA, SIM manufacturers and university researchers. Moreover we have explored context information and methods on how the Future SIM could retrieve contextual data. The goal was to get an in-depth understanding of the potential of the Future SIM.

[2] http://wiki.unik.no/index.php

- ID Management Study: We further explore the concept of identities and we identify some implementation in the smart card and SIM card, besides industry standards around it. By that, we managed to get an overview of the handling of identities in the SIM and in other platforms, besides its relation with trust and context.
- Trust Model: We assess the concept of trust and variations around trust model in order to identify possible scenarios and cases where the capabilities of the Future SIM could be beneficial. We focused on reputation based models, as the policy based ones are already vastly explored with the current smart cards and SIMs.
- Trusted Service Design and Application: After that long background research, we describe possible situations where the Future SIM can act as device to enhance trust. Additionally, we choose one of those scenarios and implement a prototype out of it, and make some experiments to evaluate the trust enhancements.

Besides that we allocated some time for writing the thesis report. In fact we did not allocate much time for it, because the whole writing process was started since the beginning in the form of structured texts in the thesis Wiki. The time division of the mentioned parts can be seen in the Figure 1.

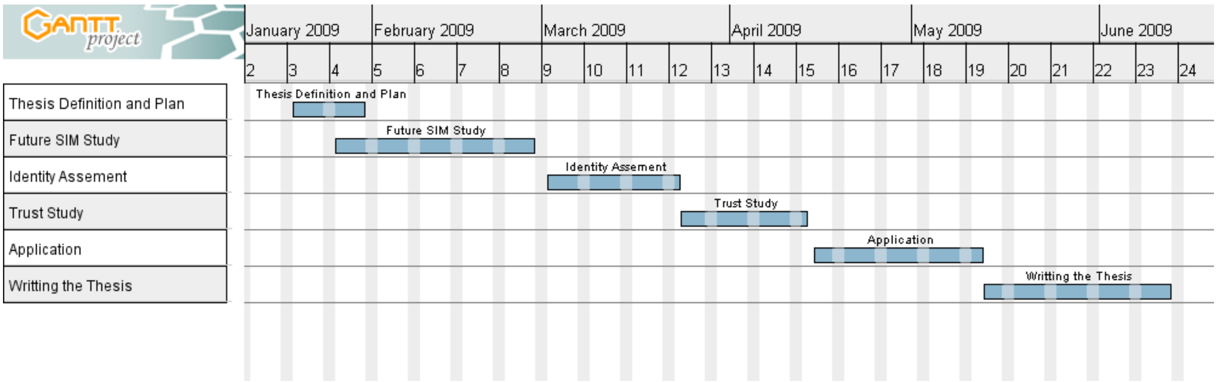


Figure 1: Thesis Project Gantt Chart

We have expanded the mentioned parts in sub-tasks and we have defined the deliverables for each sub-tasks and part. Those deliverables are represented in this report as the chapters and theirs sections.

3. Smart Cards

This chapter introduces some basic concepts of the smart card and it access the security around it, as one of the main motivations of using the SIM card as a trust platform is its security.

3.1. Smart Card Fundamentals

The introduction of the smart cards in the market started with the telephone memory cards, where the card consisted only in a non re-writable memory card with some processing logic to decrease credit from the card each time it was used (Rankl & Effing, 2004). The purpose was to protect against data manipulation of the credits in the card. Those memory cards are mainly suitable for pre-paid services or for identification services where the data is written just once in the card and whose logic is completely located in the back-end system.

The smart cards were introduced in the bank industry as a component to offer more security than the already existing magnetic stripe and also as a possibility to enable off-line payments, saving costs with data communications. In the interim, they started to be used in the telecommunication systems (with the name of SIM cards) to provide a secure authentication for the mobile towards the network. In both cases, more powerful smart cards were used, this time they included a microprocessor capable of much more advanced logic operations.

Due to this huge penetration of smart cards in different sectors, it is becoming more and more common that a regular person carries more than one smart card in his wallet. People can carry one in their mobile, one for office access, one as their national identity card, another as their bank card, etc. With the growing of usage of multi-application cards, it is becoming possible to carry just one card for all those services.

3.1.1. Smart Card Hardware

The smart cards are used to securely store data controlled by a secure Operational System (O.S.) or secure logic. It allows confidential information to be written in it and to be never read outside of the card, just processed internally by the chip's CPU. Both the hardware and the software, either the smart card operational system or applications over it, can be used to restrict the access of the data.

The smart card has 6 or 8 contacts (for the pin out see the Figure 2):

- VCC: Power supply input.
- VPP: Programming voltage input, whose use was deprecated and now is reserved for the Single Wire Protocol (SWP).
- RST: Reset.
- CLK: Clock or timing signal.
- GND: Ground.
- I/O: Input or Output for serial data.
- The 2 optional pins AUX1 and AUX2: Which were reserved for future use in accord to the ISO 7816-2 and have been allocated to the USB interface.

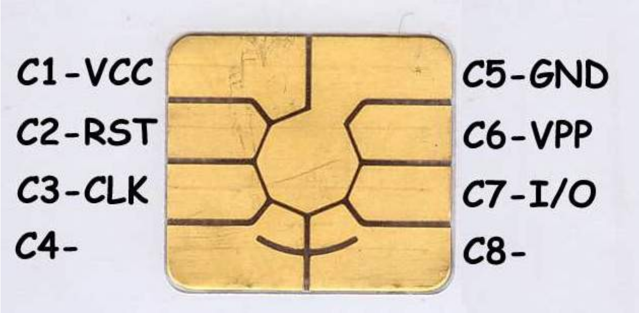


Figure 2: Smart Card Pin out[3]

As mentioned before, the smart cards hardware may consist of a microprocessor with the I/O interface, memory and CPU. The first smart card microcontrollers had CISC architecture and they were 8-bit and 16-bit controllers based on the Intel 8051 architecture or Motorola 6805. The new high-end smart cards are being developed with powerful 32-bit ARM processors. An example of its architecture can be seen in Figure 3.

The Smart Card memory is traditionally divided in ROM, RAM and EEPROM, although EPROM and Flash memory can be used as well. How much of each memory depends on the implementation, but, in general, the type of data stored in each memory is the following:

- ROM: Operational System besides some test and diagnosis functions.

[3] From http://en.wikipedia.org/wiki/File:SMARTPINOUT.jpg

- EEPROM: Variable application data such as the file system, key and parts of the operational system (called Softmasks).
- RAM: Temporary data.

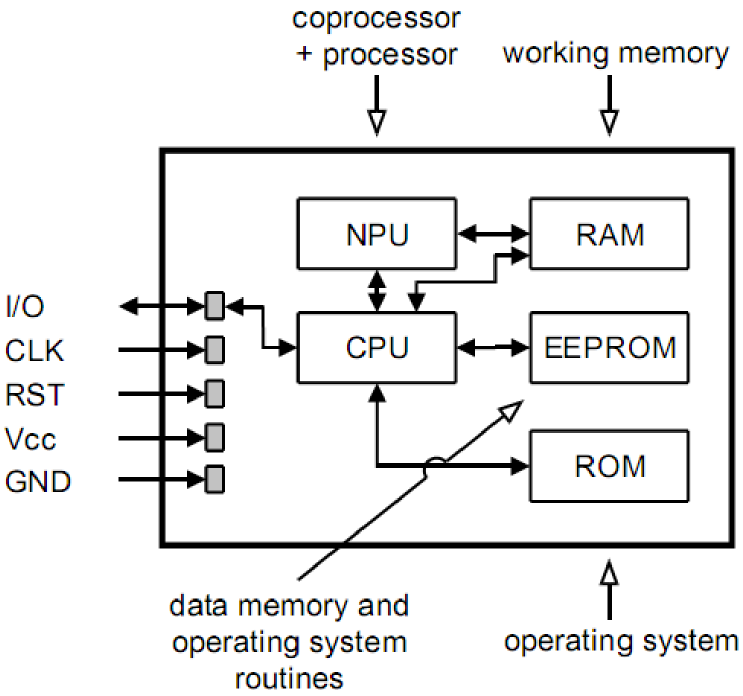


Figure 3: Typical Smart Card Architecture, from (Rankl & Effing, 2004)

The ROM is programmed during the chip manufacturing process, while the EEPROM is recorded during the personalization phase. However, it is possible to modify data in the EEPROM after the card has been personalized and issued. This enables the card content to be updated after it has been issued.

The Smart Card may have some supplementary hardware such as:

- UARTs (Universal Asynchronous Receiver-Transmitter): In order to convert parallel data transmission from serial data transmission and boost the speed communication of the Smart Card. By default smart cards only have a serial connection for both input and output of data (Rankl & Effing, 2004).
- Internal Clock Multipliers: In order to go over the 5MHz external clock restriction imposed by the standards and be able to run faster some cryptographic algorithms.

- Direct Memory Access (DMA): In order to allow copy and exchange of data at higher speeds.
- Memory Management Unit (MMU): For monitoring the memory boundaries of an application while it is running. This is used to encapsulate the applications in the smart card, and to prevent that the application exceeds the memory that was allocated to it by the Operational System.
- Cyclic Redundancy Checks (CRC) calculator units: Once the CRCs are often used by the smart cards to provide data integrity.
- Random Number Generator (RNG) units: For generating random numbers which should be immune to external source inputs.
- Coprocessors: Which are tailored for different requirements. The most common ones are usually developed for performing faster symmetric and/or asymmetric cryptographic calculations.

Besides that, there have been some initiatives to include sensors and communication interfaces on the smart cards. Some examples: the SIM card manufacturers Oberthur and Sagem Orga have announced the integration of both an accelerometer and a GPS module respectively; Telenor have announced the integration of a wireless LAN module; ETSI has standardized the integration of SIM cards with NFC modules.

3.1.2. Operational System

In the first smart cards, there was no O.S, but only a collection of libraries in the ROM. Today, there are more than thousand companies producing different smart card O.Ss, either application specific ones or general-purposes systems (Rankl & Effing, 2004).

The primary tasks of the smart card O.S. are: transfer data to and from the card, control the execution of commands, manage files, manage and execute cryptographic algorithms and program code.

All the smart card O.S. must be somehow interoperable, since they must follow the smart card standards (the ISO/IEC 708176), besides possibly GSM standards, EMV specifications and so on. They do not need to implement all the smart card standards, but they need to implement the ones related to theirs smart card applications. The standards make sure the same command sets, data structures and capabilities are compatible between different smart card vendors.

There are high requirements towards the security of those Operational Systems, due to their critical position in the smart cards. The top priority in their design is to secure the execution of the applications and the access to the data. The average size of the operational systems for smart cards is around 64kb, but the new smart cards trends are pushing this value to become much higher.

The O.S. is mainly written in the ROM, not allowing any changes in it after the chip has been programmed. Because of that, errors in the Operational System usually require chip recalls, although some errors may be corrected with softmasks, an O.S. extension in the EEPROM. This error-proof design need forces the smart card O.S. producers to spend a lot of time on testing and quality assurance.

For security reasons, the smart card O.S. must be closely coupled to the hardware of the microcontroller used. In this way, the software can be designed to counter some weakness from the hardware, as it will be further discussed in the smart card tamper-proofness section.

There are testing efforts to assure and evaluate the security level of the smart card O.S. according to the ITSEC criteria and Common Criteria[4]. But due to the cost and time needed for those security evaluations, a great range of smart card O.Ss ends up achieving only levels E4 (the Starcos[5] for example), even though some implementations such as some of the Multos[6] versions have achieved certification E6 (France-Massey, 2005).

Some other security aspects in the O.S rely on their layering composition and hierarchical file management system, which greatly control the access to O.S. routines and to the files. There are a few memory management policies such as Write Once, Read Multiple times (WORM), Last In, First Out (LIFO), best fit, defragmentation and the garbage-collection method.

[4] http://www.commoncriteriaportal.org/

[5] http://www.gdburti.com.br/brochuras/STARCOSSSPKDI.pdf

[6] http://www.multos.com/

The operations under the smart cards must always be designed as Atomic Operations. They either are completely performed or they fail. This is a very important aspect, once the card can be physically removed at any time and this should not result in a partial operation (such as paying a bill but not reducing the money from the bank account).

The Smart Cards Operating Systems, such as the Multos or the ones that supports *JavaCard*[7], allows third parties to load their own program code in the cards. This contributed to the creation of a programming API for third parties, which in several smart cards corresponds to the *JavaCard* API. Those APIs provide essential functions of the O.S. to the applications, such as calls to cryptographic functions, access to file manager, data transmission and others.

3.1.2.1. JavaCard

By the time this thesis is being written the new *JavaCard 3.0* specifications have been released but not yet deployed in the market. The latest versions deployed of the virtual machine are the 2.2 and 2.2.2

The *JavaCard* Virtual Machine 2.2.2 is a 16-bit virtual machine, and much more limited than the virtual machine of the Java Standard Edition. It has fewer data types, less instructions, no multi-threading capability and a restricted API. The Garbage Collector (which corresponds to a common Java feature of detection and freeing of unused or inaccessible allocated memory) has appeared in the version 2.2, but as an optional feature.

Still, the *JavaCard* is compatible with the existing standards, such as the ISO7816, EMV, the GSM ones and the Global Platform. The Java Virtual Machine enables interoperability between different smart card vendors, once applets developed within the *JavaCard* API will run on any *JavaCard* virtual machine. It also enables multiple applications to co-exist in a secure environment by isolating them and just allowing mutual access through shareable interfaces.

The *JavaCard* introduces the Java language as the programming language and therefore provide a more secure execution environment. It adds the usage of exceptions to the applets

[7] http://java.sun.com/*JavaCard*/

and prevents programming tricks that may lead to pointers to parts of the memory outside of the allocated boundaries. Not to mention that it makes the application more re-usable and that there already a few IDEs to develop in Java.

A great contribution of the *JavaCard* as a smart card platform resides in bringing interoperability to the smart card applications and to move the smart card development from a proprietary and closed platform to an open API that can theoretically be used by anyone to develop their own *JavaCard* applications. As seen with the PC softwares, the opening of the API's can greatly contribute to development of diverse and numerous applications.

3.1.3. Standards

There are several smart card standards due to its application in several fields of the industry. The fundamental standard for their specification is the ISO7816 and its chapters. The ISO7816 describes the standards for the smart cards from the electrical connections to the basic smart cards Application Protocol Data Unit (APDU) messages exchanged between the cards and the readers.

While the general commands are described in the ISO 7816 and the application management is described in the Open Platform, the credit and debit cards commands are described in the EMV 2000 standards; the electronic purse specific commands are specified in the Common Electronic Purse Specifications (CEPS) and the EN 1546; and the telecommunication specific APDUs are defined in the 3GPP and ETSI standards. In fact the 3GPP and ETSI standards specify much more then the APDU for the telecommunication smart card, but also physical characteristics and high level protocols.

The Visa Open Platform, renamed as Global Platform when the specifications became a standard, defines an interface inside the Smart Card O.S. in order to manage (load, install, delete) applications. It is an O.S. independent standard and the de-facto specification for loading *JavaCard* applets.

The Global Platform defines the Card Manager, which is the main component in the smart card O.S., and the Security Domains (SD). Those last ones provide keys and cryptographic services for applications that are independent from the card issuer. The SD’s correspond to one of the smart card components that enable the multi-application smart cards. By having different security domains it is possible to securely isolate the different applications and their own keys.

3.1.4. Communication

The communication with the card is always initiated by the terminal. Even, the proactive commands specified in the GSM telecom standards are based on that master-slave scheme. The communication is serial and asynchronous, relying on the usage of synchronization bits and parity bits. As mentioned before, UARTs can be used to boost the communication.

There are a few transmission protocols, but the most predominant are the T=0 and T=1. The USB protocol is a new amendment to the ISO 7816-3. It requires some special hardware components and it requires the usage of the pins AUX1 and AUX2, providing a much faster transmission rate. We will discuss more about the USB High-speed protocol later on in this thesis.

Logical channels multiplex the single physical communication channel between the mobile phone or other readers and the SIM Card. The channel number is specified by the CLA byte on the request APDU. Previously the SIM was limited to up to 4 logical channels (Rankl & Effing, 2004), but this limitation has been raised to up to 20 in *JavaCard 2.2.2* (Sun Microsystems, Inc). The logical channels are the key to the implementation of multi-application cards as they allow application to communicate with the reader in parallel. Moreover, with the memory capacity raise introduced by the high-density cards, it becomes easier to overcome the high memory requirements of having several logical channels open.

3.1.5. Applications

The smart cards are being used in several domains, but the main ones are: the telephony and telecommunication applications, financial applications, user identification, health and transportation.

In the telephony and telecom area, we have both the pre-paid telephone cards and the SIM and USIM cards that are used for authenticating the user towards the network. Moreover, smart cards are also used for the decryption of broadcast TV and multimedia Digital Rights Management (DRM).

The financial applications in the smart cards include debit, credit card and electronic purses or charge cards. Often they also carry loyalty applications that allow the retailer to obtain additional information about the customer purchasing habits, and rewards the user based on the usage of the card. There are some applications in the transport area, mainly for ticketing, parking and toll automation, where the smart card plays a role similar to an electronic purse but safely storing and sharing the ticket to the service.

The majority of the health smart card applications are used for verifying the user and his health insurance plan, (Hendry, 2001). However, they can also be used for storing and controlling the access to medical records and prescriptions.

Smart cards can be employed in physical and logical access control, by representing the user identity, or as a national or local identity. In fact some of the cards that are used to give user's access to the computer or system can be used to encrypt and decrypt data in a VPN or similar. It is worthy to say that with the multi-application smart card it is possible to have all the mentioned applications in the same card.

3.2. Smart Card Security

Security is present in all the parts of the smart card production, from the production and development processes to the design or hardware and software (applications and operational system). In order not to extend much into the topic we will discuss in this section only the key and application management process in the card, and some of the features that makes it a tamper-proof device.

3.2.1. Keys

The keys present in the smart cards are in general derived from a master key, using unique information on the smart card, such as its serial number. This key generation hierarchy exists in order to minimize the consequences of a compromised key, and following this same principle, different key are used by each algorithm and application. The key hierarchies, diversification algorithm, key storages and procedures are defined by the key management policies followed by the smart card manufacturer and the key owners.

The keys in the smart cards are identified by:

- Theirs purpose and cryptographic algorithm. By that, the O.S. assures that a key will not be misused. For example, an encryption key will not be used to perform authentication and vice-versa.
- Theirs version and theirs key identifier. This makes it easier to switch keys globally in the system, manage updates, etc.

The keys may be permanently stored on the SIM card, or they can be initially personalized but updated in course of the smart card life, or they may be dynamic keys which have different values each session.

The authentication mechanism with the smart cards is always based on the challenge-response method, assuring that the key does not leave the card. It is common to have a retry counter for the authentication keys. When the number of consecutive failed attempts to authenticate reaches a retry threshold, the card or key is blocked. It is also possible to disable and enable keys through the APDU when having the necessary permissions. The key permissions and scope are usually defined in the Security Domains ruled by the Global Platform.

3.2.2. Global Platform

The Global Platform is a cross-industry organization working towards the maintenance and promotion of multi-application smart cards standards. The organization encompasses members around 50 institutions from several different industries such as financial institutions, telecommunication providers, smart card and terminal manufacturers, software developers, etc. (Markantonakis & Mayes, 2003)

The relations between the Global Platform and ETSI were initiated in 1999, to standardize the OTA application download and management (Bernabeu, 2007). Theirs specifications became the de-facto standard for applet management in the *JavaCard* platform.

Two of the main components of the Global Platform standards related to the cards are the Security Domains, which can be seen as special types of applications, and the Card Manager. In the new version of the standards, the Card Manager is divided in three parts: the Issuer Security Domain, the Global Platform Environment (OPEN) and the Cardholder Verification Methods (CVM).

The card manager represents the card issuer and it is the main responsible for the security in the card. It is the entity that dispatches the APDUs and selects applications inside the card. It performs secure memory management, controls the content management (installation, selection and removal of applications in the card) and controls the card's life cycle, which is stored in the card Registry.

In the other hand, the security domain represents a secured region under the control of the security domain owner, either the card Issuer or an application provider. A security domain is isolated from the other domains. Only the issuer security domain, which is in control of the telecom operator in the case of the SIM cards, can interfere on the others security domains. However, this interference is restricted to either the creation or removal of a domain. That means that the issuer could never change a key in another domain, at maximum remove that domain. As a result, we have an architecture where an application provider that owns a security domain is sure not to have his security tampered not even by the issuer. But the issuer, as the provider of the card, has powers to remove the security domain of an application provider in case this provider shows himself to be fraudulent or malicious.

The security domains allow the domain owner to provide cryptographic services such as key management, encryption, decryption, digital signature generation and verification. Those services can be shared with other applications, through mechanisms such as shareable interfaces and Java RMI. The security domains are also responsible for verifying the Load File Data Block Signature, called Data Authentication Pattern (DAP), for operations that require loading a file under its security domain.

Each application is linked to a security domain and they can access their cryptographic services during personalization and runtime. This access is done through the Open Platform API and it is the Card Manager that is responsible to dispatch the APDUs for the authentication to the corresponding Security Domain, or the Issuer Security Domain, if not specified. Figure 4 shows a block architecture of the Open Platform.

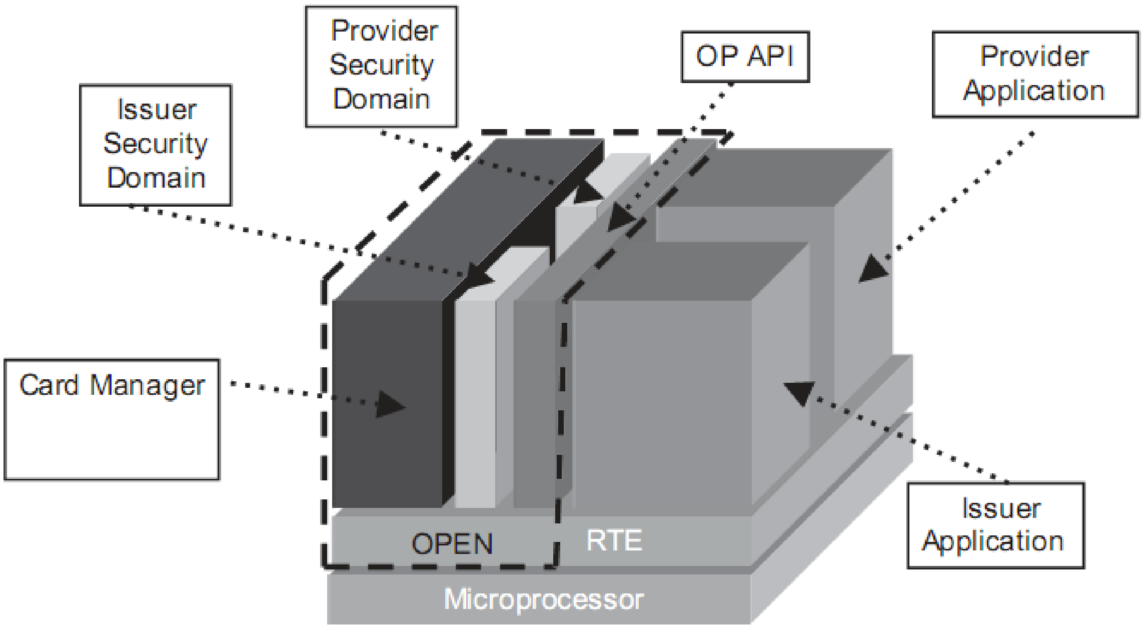


Figure 4: Open Platform Architecture, from (Markantonakis & Mayes, 2003)

The application is initially associated with the Security Domain which loads it, but it can be extradited to another security domain during the loading process or afterwards. Therefore, there are two approaches for the smart card to host a secure application from a third party service provider.

In the first approach, the application has its security domain created during the personalization phase of the card, before it leaves the factory. The domain keys are created and personalized in the secure environment of the factory at that phase. In that way, the master key, which generated each card key, can be managed only inside the Hardware Security Module (HSM) of the production site and without the disclosure of the key. Due to the fact that the operator does not know the key values at any point, this option can be considered more secure for the content provider. Once the keys have already been created, their value can be later updated by the application provider, but their characteristics (size, algorithm used) can not be changed. As mentioned before, the operator can not update or change those keys. He could at most remove the security domain, which would avoid the application to run but would not tamper its behavior.

The second solution involves the creation of the service provider security domain via OTA, or other post-personalization method, targeting the card manager and using the issuer domain to put a temporary key. Then, the temporary master key is transferred to the Service Provider which can use it to update the card. This case leaves more flexibility, once the choices of defining the security domain are taken after the card has been issued. Moreover, it can also target the legacy cards already on the market. In the other hand, the service provider must trust the issuer for having the key information in the beginning of the process. This approach has to bear with constraints as: the secure transport of the messages (as OTA encryption) and the availability of the target cards. The new release of the Global Platform Standard, the 2.2, specifies that the Card Content Management can be performed through asymmetric cryptography and Public Key Infrastructure (PKI).

In summary, the Global Platform specifies methods so multi-application cards can be securely developed and which both the issuer and the application provider are protected from each other. It enables the smart card to work as a key selector or framework for independent secure applications.

3.2.3. Tamper Proofness

A great number of different kinds of systems are now relying on the security features of the smart cards. Their tamper-proof protections have become more and more important for their establishment in the market. This high demand of security has triggered a lot of effort from the scientific community to attack and evaluate the security of the smart cards. In the other side, the manufacturers keep working on counter measures to enhance the security of the micro controllers against newly developed attacks. This battle, between finding vulnerabilities and fixing them, has been evolving fast, and 10 year old cards can usually be tampered, as show in (Anderson & Kuhn, 1996). There are mainly two types of hardware attacks: the invasive attacks and the non-invasive ones (Renaudin, Bouesse, Proust, Tual, Sourgen, & .Germain, 2004).

The Invasive Attacks are done in the physical level. They start with the violation of the tamper-resistance on the chip surface, then reverse engineering and probing the chip in order to observe and manipulate the communication with the Integrated Circuit (IC).

The non-invasive attacks exploit the hardware implementation weaknesses of some algorithms and apply techniques such as side-channel attacks to discover the secret without breaking into the physical protections of the card. Those two forms of attacks can be combined in what is called semi-invasive attacks.

3.2.3.1. Invasive Attacks

For performing an invasive attack, first the attacker must obtain physical access to the chip. In general it is quite simple to remove the module from the card. Then, the epoxy resin must be dissolved from the chip, which requires fuming nitric acid and some heating source. After those steps, the silicon is exposed and it can be attacked.

Most chips also have a passivation layer for protecting from oxidation and other chemical process. This layer can be removed through the usage of dry etching with hydrogen fluoride, or microprobing needles using ultrasonic vibration, or laser cutter microscopes, or an electron beam tester. It requires a far more advanced range of equipment available and can not be easily performed by amateur attackers. Moreover, after the passivation removal, the chip is exposed to oxidation which can destroy it quickly if the environment is not prepared.

A protection for the attacks to the passivation layer consists in adding a sensor circuit to perform measurements determining if the passivation layer is still present. If it is damaged or not present, the circuit can shut down or block the card.

Memory buses and the layout can then be tapped by probing the chip with the appropriate equipment. Microscope images can be taken with different layers of the chip in different colors. Countermeasures for those are based on:

- Reducing the size of the circuits, to make it harder to analyze them.
- Adding light sensors to avoid intentionally EEPROM erases, once EEPROM can be erased with UV light.

- Adding integrity checksums to the EEPROM memory chunks help to counter erase attacks.
- Protecting the design of the chip by adding dummy structures to confuse attackers.
- Manufacturing both the buses and ROM in the lower layers of the chip.
- Scrambling the buses and the memory.
- Memory encryption, particularly for regions used for storing keys.
- Adding sensors for checking interruptions, temperature, voltage, short circuits and irregular frequencies.

When applying sensors to detecting the mentioned attacks, it must be considered that the sensors need external power supply to react against the intrusions and they can be destroyed when the power is off. Therefore, the presence of the sensors must be periodically checked.

Since the RAM memory loses its content when powered off, in order to attack it, it is necessary to cool it down to a temperature around -60 degrees Celsius so the data remains on the memory. This is a difficult attack which involves the removal of the metallization layers underneath the passivation layer and needs voltage-sensitive scanning electron microscopes, but it can be countered by encryption. Moreover, the session keys should be usually immediately erased from the RAM after usage.

3.2.3.2. Non-Invasive Attacks

Non-Invasive Attacks corresponds to logical attacks, side channel analysis and fault injections.

Side Channel Analysis uses the timing, power consumption and electromagnetic information leaked when the cryptographic algorithm or other operation is performed.

The power consumption can be analyzed through Simple Power Analysis (SPA) or Differential Power Analysis (DPA). The SPA uses the power consumption values to gain knowledge about the data and instructions processed. While the DPA uses the statistic sampling and analysis of correlation between the power measurements to draw conclusions on the key information. There are a few techniques to make it harder to do a power analysis. Some of them are:

- Add noise generators to insert random instructions to mislead the attacker.
- Randomize the order of some operations.
- Use a modified processor that draws constant currents.
- Use only machine instructions with similar power consumptions.

The main constraint of the mentioned techniques is that they end up raising the amount of power consumed.

Timing attacks take advantage of cryptographic algorithms that takes more time for different keys or different plaintexts. Based on the knowledge of the time correlations of the algorithms and some computations, it is possible to discover the keys. This technique is usually countered by noise-free cryptographic implementations, whose time required for encryption or decryption is the same for any input. Another mean to limit those attacks is implementing by retry counters that limit the number of attempts.

It is also possible to perform a side channel attack by measuring magnetic fields dimensions and strengths using Superconducting Quantum Interference Devices (SQUIDs) to leak information. However, this is technically very difficult because the tracks are in general one over the other in the chip.

Furthermore, fault injection attacks corresponds to the introduction of abnormal inputs, such as power or clock glitches, in order to make the microcontroller perform different instructions or to corrupt a verification check. This method can be used for applying Differential Fault Analysis (DFA) which exploits computational errors in cryptosystem using properties of modular arithmetic to find the keys. As an example, in the DES and triple-DES algorithms, just around 200 single-flipped bits are needed to find the secret key (Karri, Wu, Mishra, & Kim, 2001).

Fault injections can be prevented by adding the already mentioned sensors to detect light, voltage, power changes and electromagnetic interference. Besides that, data checksums and some defensive computing on checking the validity of the data can be used against those attacks.

3.2.3.3. Software Attacks

There are also attacks performed on the software in the microcontroller. Software attacks can target poorly implemented logic, cryptographic algorithms that have been broken or exploit vulnerabilities such as buffer overflow. Some attacks can combine both software and hardware measures, such as the timing and power analysis attacks. Thus, it is very important that the hardware and software are closely coupled. In practice, it is common for the smart cards O.S. Producers to use different masks for different chips and different hardware manufacturers (Rankl & Effing, 2004).

The smart card software development usually follow security principles including auditions, extensive testing, prohibition of undocumented features, production in access-restricted environments, impossibility of switching back from user mode to test mode and having master keys stored in highly security device such as a HSMs.

The operational systems and applications also employ security aspects such as application isolation, data borders protection, usage of atomic transactions, integrity checksums for important memory data, layer separation and strict access controls.

The software and operational system security features are even more important in the context of multi-application smart card since interoperability and inter application isolation are key factors. But, as already mentioned, the standards of the Global Platform provide standard mechanisms for firewalling and protecting the distinct applications on the card.

Besides the hardware and software attacks to the hardware, social attacks can happen by targeting the people that works with the Smart Cards. Those attacks can be fought by employing strong security policies and constant auditing. Moreover, the usage of open standards and third-party auditors improves the security against those attacks as well.

The conclusion of the tamper proofness of the smart cards can be summarized, as (Markantonakis K. , Mayes, Tunstall, Sauveron, & Piper, 2007) says, in a cyclic process where new attacks are developed against the algorithms, standards, API, hardware and, then, countermeasures are proposed and applied. This process forces the smart card industry to be securely ahead of the attacks that can be performed at lower cost than the value of the information stored. This also influences the establishment a reasonable validity period for the card, once it is natural that after some years security schemes tend to be broken.

The wide-spread usage of smart cards for sensitive applications, such as e-passports, bank cards and authentication cards, proves the tamper proofness of those devices and point them as a reliable security framework for several applications.

4. SIM Cards

In this chapter we move the discussion from the smart card to specifically the SIM cards. We focus on its low level and high level communication protocols and we assess the new enhancements that are being developed in it.

4.1. SIM Fundamentals

The Smart card microcontroller, together with the electronic circuit and operational system is referred as the Integrated Circuit Card (ICC). However, in the 2G mobile telecommunication context, it got the name of Subscriber Identity Module. In fact the name addressed both the logical and physical smart card entity. With the introduction of multi-application cards, the SIM started to refer to the SIM application, the logical component; which could actually be implemented over another secure element different from the smart card.

Similarly, the Universal Integrated Circuit Card (UICC) appeared as the physical card containing the operational system targeting the UMTS mobile networks, and the USIM appeared as the logical application for the UMTS in the UICC. The UICC may contain the SIM, USIM and several other applications. However it is common in the literature to call the conjunct USIM and UICC as USIM and the conjunct SIM + UICC or SIM + ICC as merely SIM (Rankl & Effing, 2004). In this thesis we will often refer to the UICC +SIM or USIM as SIM.

Thanks to the high rate of evolution in the telecommunication market, the SIM cards have leaded the advances of smart card functions and memory capacities. Their main function is to prove the authenticity of the mobile station in respect to the network. But as we will present, theirs functions were expanded to become the secure element for several applications, to act as an identity and profiling device for the user, and, act as a communication channel between its applications and supported networks.

The SIM is described by the GSM standards that were managed by the GSM Association. Now the GSM standards have been incorporated into the 3GPP standards managed by 3GPP and ETSI. In general, the ETSI/GSM standards are very precise and define unambiguously the SIM interfaces, leading to compatible and interoperable implementations. Some important GSM standards are:

- GSM 11.11: Which describes the interface between the mobile equipment and the SIM card, besides the SIM file system.
- GSM 02.17: Which describes the SIM functional characteristics.
- GSM 11.14: Which describes the SIM Application toolkit (SAT).
- GSM 02.48 and 03.48: Which describes the secure data transmission via OTA (Over-the-Air), the Remote Applet Management (RAM) and Remote File Management (RFM).

The *JavaCard* has become the natural platform for the SIM O.S., and now, some of the 3GPP and ETSI specifications address the *JavaCard* specifically. The GSM 03.19, for example, details the implementation of the *JavaCard* API.

Besides its authentication capability, the SIM also stores data. This data ranges from the agenda, short-messages (SMS) to network settings such as roaming lists, the international mobile station identifier (IMSI), and the mobile station international subscriber dial number (MSISDN). This data is stored in files in the SIM application. The SIM provides a hierarchical file system with both directories and files. The files may have record, cyclic record and linear structure. Read and write access, creation and deletion of files and directories are ruled by detailed access conditions based both in the action to be done and the authentication key used.

The SIM, as a smart card, can support symmetric and asymmetric cryptography, storage of certificates and perform digital signatures. The network authentication uses symmetric cryptography. Until the 2.5G, this authentication is not mutual, just the SIM card authenticate itself to the network. Consequently, it is possible to eavesdrop calls with the suitable piece of equipment, such as an IMSI catcher, without knowing the key in the card. The voice traffic is encrypted, but the encryption and decryption is not performed by the card. Instead, the key to encrypt/decrypt the voice is provided to the mobile from the SIM card. In the UMTS networks, the USIMs authenticate the network besides authenticating themselves.

4.2. Physical and Logical SIM Interfaces

The handset acts as the access channel to the UICC and ICC, once it is through the embedded handset reader that both the user commands, network command and handset applications get in contact with the SIM/USIM. This access is done through the communication protocols such as ISO 7816 or the new high speed protocols USB 2.0 or MMC.

There are also three high level standards of interaction between the mobile phone and its applications and the SIM. They are the SIM Application Toolkit, the Smart Card Web Services (SCWS) and the JSR-177 (also known as SATSA). In what concerns the communication with the Network operator, the SIM can communicate through the SMS channel, Cell Broadcast, advanced data channels such as GPRS through Bearer Independent Protocol (BIP) and recently it is being standardized the connection through the IP Channel.

Besides that, some sensors and other wireless communication interfaces have been embedded in a few SIM cards.

4.2.1. SIM Application Toolkit

The SAT was standardized in 1996 by ETSI in the GSM 11.14 standard. It enables the SIM card to access some of the mobile phone functionality such as display, keypad, SMS, tones, etc. The commands are directed from the Mobile Equipment (ME) to the SIM. But there are the proactive commands, which work on the opposite direction.

The proactive commands are based on the fact that the ME issues a polling command regularly to detect if the card is still there. Then, if the card wants to start a communication, it gives a special answer to the polling interval. This special answer will trigger a fetch APDU from the mobile to catch the proactive command. After receiving and interpreting the command, the ME sends back a terminal response to the SIM. In addition, the SIM can inform the mobile about some events (such as SMS receive, call setup, SIM menu access) which it must be immediately notified when they happen.

Derived from the SAT, we have:

- <u>the Card Application Toolkit (CAT)</u>: Which defines the generic foundations for all application toolkits for smart cards in mobile telecom.
- <u>the USIM Application Toolkit (USAT)</u>: the equivalent of the SAT for the 3G.

Some examples of proactive commands that offer some of the mobile resources to the SIM via the SAT are:

- Display Text: Allows the SIM to use the mobile screen to display a text. The response to this command informs how the user reacted to the display, as if he pressed a function key or ignored it.
- Get Input/Get Inkey: Enables the SIM to prompt a message to receive some input from the user through the mobile keyboard.
- Setup Menu: Allows the SIM to provide a text menu to be displayed in the handset. The selection of one of the items in the menu triggers events that will call the SIM or an application inside the SIM.
- Select Item: Allows the SIM to pass a list of items to be displayed in the handset and receive the information of which item was chosen by the user.
- Send Short Message: Provides the SIM with the capability to send a SMS.
- Setup Call: Enables the SIM to start a call.
- Provide Local Information: Retrieves data from the handset or network about local time, network position (which cell is it in) and network cell strength.
- Set up event list: Allows the SIM to inform the handset about handset event that it wants to be notified about. Some of those events are: received call, received SMS, new location information received or if the mobile enters idle mode.
- Timer management: Gives the SIM access to set and get informed about timers.
- Channel commands: Provide commands such as Open/Close Channel, get channel status and send/receive data which are used to open communication channels in between the SIM card and the mobile, such as the BIP channel.

As it can be noticed, the SAT offers primitives for the development of somehow complex application on the SIM card. SIM card vendors and developers have managed to deploy several neat applications such as: Agenda Backup[8], Handset Configuration[9], Instant Messaging Client[10], Mobile Banking[11] and OTP Solutions[12]. The SAT applications can be programmed using the *JavaCard* API, but still there are just few developing tools for it and the SAT applets popularity has been limited due to its constrained GUI.

[8] http://www.gemalto.com/brochures/download/linqUs_phone.pdf
[9] http://www.gemalto.com/brochures/download/linqUs_device.pdf
[10] http://www.gemalto.com/brochures/download/SIMessenger.pdf
[11] http://www.gemalto.com/brochures/download/mobile_banking.pdf
[12] http://www.todos.se/downloads/Todos_inSIM.pdf

More information about the SAT commands can be both found at the GSM standards such as the GSM 11.14 or at (Guthery & Cronin, 2001).

4.2.2. JSR-177 - Security and Trust Services API

Another possibility to explore the secure features of the SIM card is through the JSR-177 Security and Trust Services API (SATSA) for Java 2 Micro Edition (J2ME). The J2ME is a mobile platform based on the Java Standard Edition platform, but adapted to embedded devices. J2ME has been deployed in several handsets in the market, but the implementation of the SATSA API is not that much spread as it is an optional API.

The JSR-177 API enables the J2ME application (known as *Midlet*) to make use of a secure element, which can be the SIM Card or another smart card, to ensure trust and security to the *Midlet* service. Thus, the *Midlet* can delegate the sensitive data storage, or secure execution and cryptographic operations to the secure element. A diagram of the communication channel between the J2ME *Midlet* and the SIM application on the card can be seen in Figure 5.

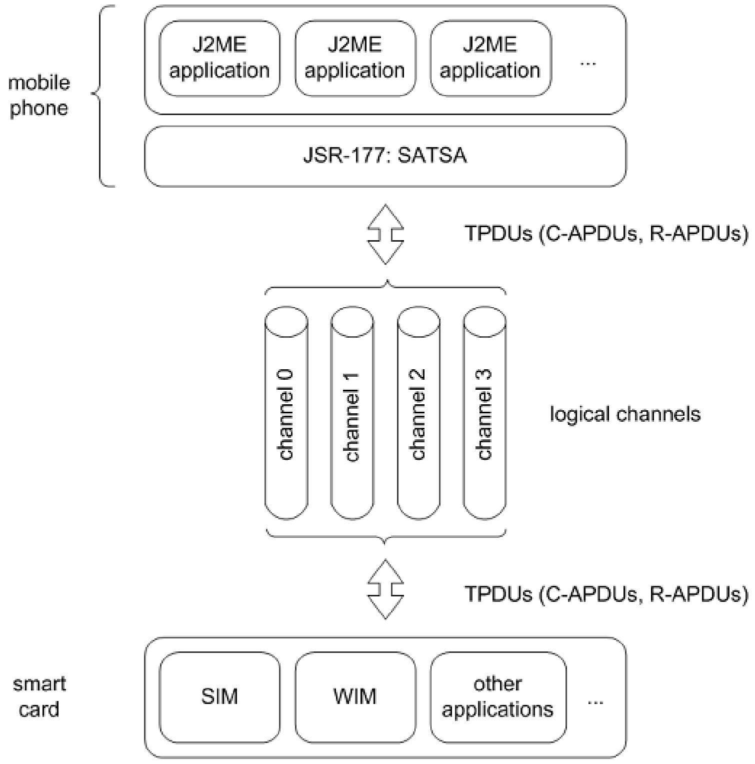


Figure 5: Path from J2ME to SIM application, from (Eisl, 2004)

The JSR-177 API mainly provides the following high-level functionalities to the *Midlets*:

- A communication API for the accessing the Smart Cards, through both APDU exchange and JCRMI, implemented by the packages SATSA-APDU and SATSA-JCRM.
- A public key cryptography API (the SATSA-PKI) for enabling the management of digital signatures, certificates and credentials.
- A cryptography API called SATSA-CRYPTO, providing basic cryptographic algorithms for encrypting, decrypting and for generating and storing keys.

The SATSA also specifies access control conditions for the *Midlet*s to access the Secure Element (SE) through both the SATSA-APDU and SATSA-JCRMI packages. The controls mainly protects the SE from malicious applications that may try to perform a denial-of-service attack or similar. The secure element can restrict its access by providing a domain root object, such as a trusted certificate or public key. Then, it will only trust the applications inside that domain, like a *Midlet* signed with a certificate that chains back to the one of the SE. Alternatively or the SE can publish its access control list in an Access Control File (ACF).
The SATSA allows two communication methods between the *Midlet* and the secure element. One of them is through the APDU protocols and the other one is through the Remote Method Invocation (RMI) protocol.

The procedure through RMI requires the implementation of the *JavaCard* RMI in the *JavaCard* operating system of the smart card. The JCRMI provides a higher abstraction interface, so the *Midlet* acts as a client requesting services to the smart card, instead of sending an APDU and getting a response. The *JavaCard* applet must start a remote object that will be accessible to a client stub object. Then, the *Midlet* application will call a high level method of that object that will be translated into APDU by the stub objective and then received by a skeleton object of the remote object. Subsequently, the skeleton object will translate back the APDU to the method call of the remote object (Oostdijk & Warnier, 2003).

The procedure through APDUs mainly consist in the *Midlet* opening a logic channel targeting an application in the *JavaCard* and, then, exchanging Command and Response APDUs with it. In the SATSA, the opening and management of the logical channel is done transparently by the API.

The SATSA offer a possibility to enhance the visual interface towards the content and applications in the SIM. It corresponds to a possibility to extend to the SIM, the J2ME rich GUI and also other features from the phone such as: accelerometers, GPS, WLAN connectivity, etc.

4.2.3. Over-the-Air

As mentioned before, the GSM 03.48 defines the specifications to allow secure end-to-end communication between the SIM card and the background OTA system. The most common transmission channel for that is the SMS. However, the OTA messages can also go through a data communication channel, such as GPRS encapsulated by the BIP.

The GSM 03.40 makes it possible to specify that the message targets the SIM application, and to specify a specific application in the SIM by targeting its Toolkit Application Reference (TAR). It is possible to break the data to be transmitted in several messages, overcoming the size restrictions of the SMS, and to have the SIM sending back a Proof of Receipt (PoR) by OTA. This enables a bi-directional communication channel between the end-system and the SIM card.

In order to secure the OTA management of the card and its messages, it is necessary to either use the keys of the Telecom Operator or to have a key exchange between Service Provider and Operator. Depending on the security level required, the messages can be encrypted and/or authenticated.

The SIM micro-browser defines a way to browse content in the Operator’s network through the OTA communication. It usually consists of a menu on the SIM just like a URL list. Then, by choosing one of items, the user can exchange messages with the service provider through OTA, enhanced by the SAT browser. However this requires that the SAT browser knows how to format the message, the address of the OTA platform that will work as a web server for the micro-browsing, and to know the capabilities of this last one.

One of the most popular SIM browsing solutions is the WIB Browser which was invented by SmartTrust[13] and corresponds to a closed browser that talks a variant from WML. Another option is the S@T Browsers defined by the S@T alliance and whose message format is S@TML. Both browsing can be protected with encryption keys or signed with digital signatures, as they exchange regular OTA messages. Both microbrowsing are not mutually compatible and not standardized. In fact, ETSI has published the specifications of a browser called the USAT interpreter.

The microbrowsing is a solution to address the large amount of low-end users whose only interface towards data remains in the SMS channel. However, it is limited to the content providers which have their menus published on the SIM and possibly have microbrowsing keys on the card. The application browser actions and interface are also somehow limited by the SAT.

4.2.4. Cell Broadcast

Another option of targeting the mobile devices is through Cell Broadcast messages. Those messages target all the devices compatible with cell broadcast, whose option of cell broadcast is turned on, at a cellular cell area. There is no information about the receipt of the messages from the people in the cell.

Since a dedicated network infrastructure is used to carry these Cell Broadcast messages, the broadcast is not affected by the peak traffic of voice or SMS. And, it does not interfere on those services as well. The SIM Toolkit has mechanisms to be triggered by the Cell Broadcast Messages, making it a possible interface for broadcasting messages directly to applications in all the SIM cards in the area.

Several services can be addressed through this channel such as mass location based services based on cell location, mobile advertisement, communication of information inside events or emergency notification. (Cell Broadcast Forum, 2002) provides a comprehensive list of services that can be addressed through Cell Broadcast.

[13] http://www.smarttrust.com/

4.2.5. Bearer Independent Protocol

The Bearer Independent Protocol enables the USIM to establish a communication channel through a phone's high bandwidth channel such as GPRS, EDGE or HSDPA. Those channels drastically improves the speed of RFM and RAM operations that were so far limited to be performed through the SMS-PP download methods using the small sized SMS messages as data carrier. In any case, the platform or the SIM can choose between the data carrier, either as SMS or a high speed one, depending on the bearer availability.

The smart card application communicating over BIP can act in:

- Client mode: When it proactively opens the channel to communicate with a remote platform of known IP. Then, the handset works as a gateway translating the TCP/IP to BIP or CAT_TP to UDP/IP).
- Server mode: When the smart card is the server allowing external TCP applications to connect to it. In this case, the SCWS should be implemented so it receives the TCP requests targeting the localhost IP address in the SCWS ports (3516 and 4116).

Anyway, in order to exchange data with a server through the bearer channel, it is necessary to use a data transport protocol which could be the TCP or the CAT_TP. The CAT_TP establishes the communication directly between the SIM Card and the Server, being necessary that both SIM and Server implement the CAT_TP protocol. Below the CAT_TP, the data is transferred as UDP datagrams. In the other hand, it is possible to use the TCP protocol through the implementation on the mobile phone. The diagram of the communication between the SIM application and the server through BIP, both on CAT_TP and TCP mode is shown at .

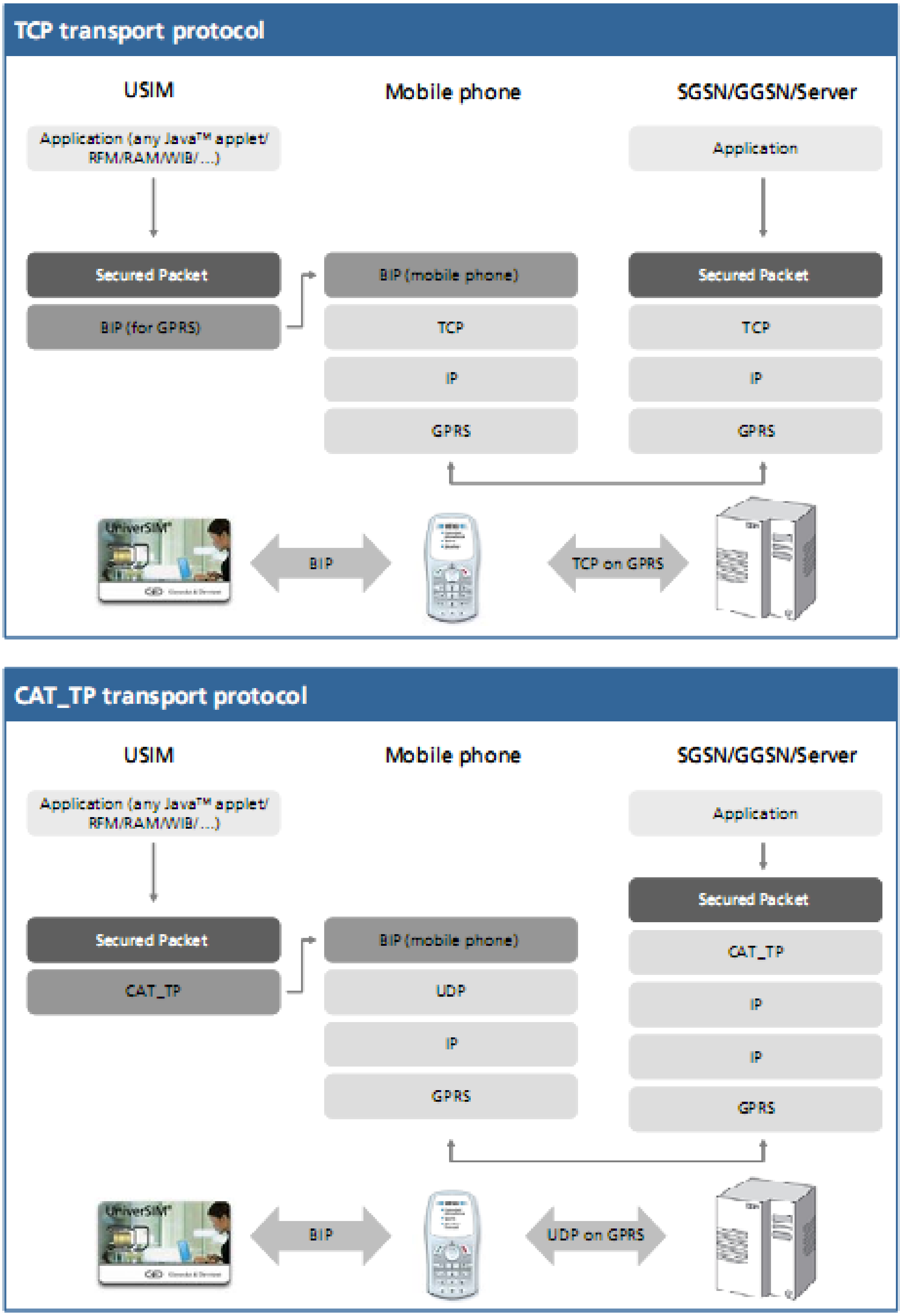


Figure 6: BIP through GPRS, from (Giesecke & Devrient, 2006)

The service to be carried through BIP can be pushed by the operator via OTA through a command to open the Bearer data channel or through an Applet or SIM menu that triggers the CAT command OPEN CHANEL. When the data communication is finished either the phone sends a "link-broken" event to the card or the card sends the proactive CLOSE_CHANNEL command to the phone.

Today, there are just few handsets that implements BIP. However, the standardization of the SCWS protocol and the High Speed USB channel should encourage the release of more handsets with BIP enabled. The SCWS promises to enhance the user interface towards the SIM and also facilitate the development by having HTTP and Java Servlet web applications together with the already existing *JavaCard* applications, as it will be mentioned later on in this thesis.

4.3. New SIM trends towards the Future SIM

We describe here the trends towards the Future SIM based on pilots, white papers, news and technical specifications around the SIM published by related institutions, smart card providers, operators, etc.

4.3.1. Physical Enhancements

The new trends around the physical characteristics on the SIM correspond mainly to an increase of memory capacity and the development of new communication interfaces towards the UICC, such as the USB and SWP interface. The picture shows the SIM and its interfaces towards the baseband controller or mobile phone and NFC controller.

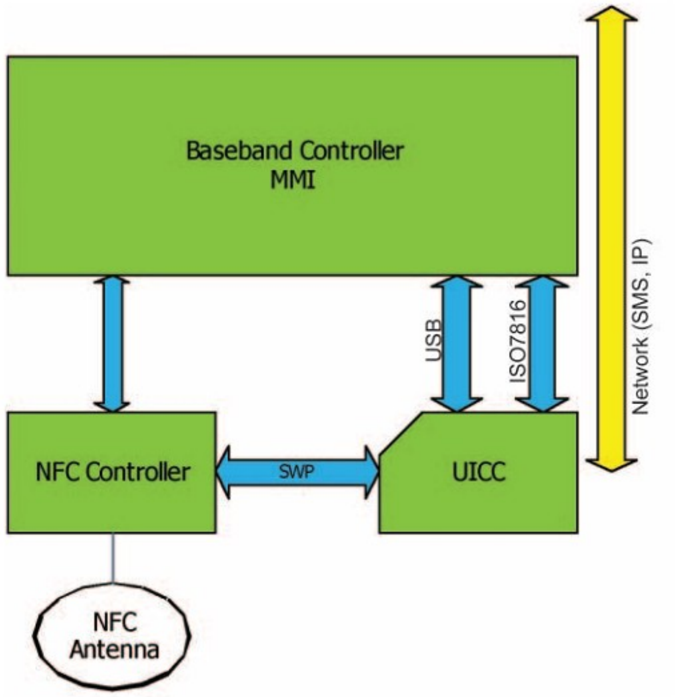


Figure 7: UICC and its hardware interfaces, from (GSM Association, 2007)

4.3.1.1. High Density SIM and USB

One of the big changes on-going in the SIM card platform is the increasing of its memory capacity. This is something that will greatly differentiate the current mass deployed SIM to the Future SIM, as it seriously expand the range of applications that can be deployed on the SIM.

The interest in increasing the SIM capacity to the range of megabytes has appeared in 2004, when the high density NAND flash technology has enabled the common size of the SIM Cards to be extended from the 64-256kb to Megabytes or even Gigabyte sizes, as announced by chip manufacturers such as Sandisk[14] and Samsung[15].

There is a trend on replacing the memory in the SIM Cards with the flash memory due to its maximum capacity and density, minimum power consumption, and a high number of write cycles. The fact that newer flash memories can store more bits per memory cell also helped to decrease the cost per bit of the Flash, making it more pricey competitive (Glass, 2000).

[14] http://www.sandisk.com/Assets/File/pdf/oem/SanDisk_Family_of_SIM_Solutions.pdf

[15] http://www.samsung.com/global/business/semiconductor/support/brochures/downloads/systemlsi/smartcardic_061107.pdf

In the Flash technology there are mainly NAND Flash memories and NOR Flash memories. The first ones have high density, medium read speed, high write speed, and an indirect or I/O-like access. Thus, they are more suitable for big data storage. The NOR Flash has lower density, high read speed, slow write speed, but a random access interface; being more suitable for storing the operational system of the smart card or application code. Nevertheless Multi Chip Packages (MCPs) and System in Package (SiP) technologies enables both memories to be combined (Toshiba America Electronic Components, Inc., 2006).

Table 1: Technical characteristics of traditional vs. high density cards, adapted from (Handschuh & Trichina, 2007)

	Typical Card	**High Density Card**
Confidential Operating system	ROM (512 KB)	CodeFlash (512 KB NOR Flash)
Application Data, Secret Keys	EEPROM (256 KB)	Emulated EEPROM (128 KB NOR FLASH)
RAM	5 KB	24, 48, 64 KB
User Data	In EEPROM	4MB to 1 GB NAND FLASH
Interface	ISO 9600 bit/s	+ USB, MMC High-speed protocols
Die Size	25 mm2 in 0,13µm technology	Less than 80 mm2 in 90nm technology

The new generation of high density SIM cards basically replaces both ROM and EEPROM by FLASH. They use NOR Flash for the information that was in ROM, some application data and keys; and use NAND Flash for user data. The Flash technology introduction also allows a great speed gain in the process to load the mask in the chip, but it imposes some security challenges, such as:

- The need to protect the proprietary and highly sensitive data (operational system and private algorithms) that were stored in the ROM. This demands memory scrambling in hardware level to protect its reading and also locking the memory with Write Protection.

- The limitations of the Flash erase, which only allow whole sectors to be erased. Thus, it is necessary the development of algorithms for EEPROM emulation and anti-tearing, protection of memory content when the card is removed in the middle of a transaction.
- The need to implement the microcontroller and the flash memory on a single die packaging in order not to have an external bus linking both of them and being an easy target for probing attacks.

This increase in the memory size faced an obstacle on the maximum data transfer of the ISO 7816 standard smartcard interface, limited in the order of 9.6Kbit/sec. This lead to the proposal of two standard interfaces for high-speed data transfer: the Universal Serial Bus (USB) and Multimedia Card (MMC) interfaces, which allow speeds of 12Mb/s and 26Mb/s respectively. The USB has the advantages of using only 2 pins, being robust and having IP support. For the MMC it counts that there are already a few handsets implementing the MMC interface, and it is cheaper and consumes less power. However, the MMC interface requires 3 connection pins of the card (Mardiks, 2005).

After a period of discussion if either the MMC or the USB would become the standardized high-speed protocols, ETSI ended up agreeing in the USB standard, which is now specified in the TS 102.600 standard. However, some chip producers, such as Samsung[16] and SanDisk[17], are producing chips compatible with both USB and MMC standards due to the unavailability of USB compatible terminals so far. It is important to mention that, for compatibility reasons, the handsets and high density SIM cards must also support the ISO 7816 transmission protocol besides any high speed protocol.

The USB protocol standardized for high speed communication in the SIM through the pins 4 and 8 was the Universal Serial Bus Inter-Chip (USB-IC), which is a variation of the USB 2.0 protocol adapted to the smart card realm.

Three USB classes are supported, but only the USB Integrated Circuits Card Driver (ICCD) is mandatory. The USB ICCD corresponds to the exchange of the APDUs over USB and APDUs to configure the transmission bulk pipes.

[16] http://www.hartware.de/press_6087.html
[17] http://www.sandisk.com/Assets/File/pdf/oem/SanDisk_Family_of_SIM_Solutions.pdf

The USB Ethernet Emulation Mode (EEM) is optional, but it can bring great value for supporting IP applications over the USB interface. The EEM is the encapsulation of Ethernet frames in the USB bus, providing TCP/IP connectivity. It is ideal for the Smart Card Web Services interface and the IPSIM.

The support for the Mass Storage Bulk (MSB) mode is also optional but mandatory for accessing mass storage. This USB class gives an interface to several storage devices without any specific file systems to be implemented, as it only provides a simple interface to read and write sectors.

The high-density SIM cards enable more complex applications as there is more space in the SIM and no speed limitation on the communication interface. Key storage, profiling and sensitive information can be largely stored in the SIM. Even, if some data is still too large, it could be stored in the mobile and quickly accessed through the USB high-speed interface.

4.3.1.2. Near Field Communication

The SIM Card connectivity is being further enhanced by the standardization of the Single Wire Protocol. The SWP, specified in the TS 102 613, defines the physical communication interface between the SIM Card and a contactless frontend (CLF).

The SWP establishes a master slave communication where the CLF is the master and the UICC is the slave. The communication is full duplex on a single pin, with the signal sent by the master being transmitted as a digital modulation in the voltage domain and the one of the slave in the current domain. The I/O connection between the two parts is through the pin 6 of the smart card. ETSI has also standardized the Host Controller Interface (HCI) in the TS 102 622 document, providing a low-level software API standard on top of the SWP. The SWP and HCI together enable to host secure application on the SIM and communicate through the external world through the contactless frontend.

This contactless frontend is being represented in practice by a NFC (Near Field Communication) chip. The NFC is a standard defined by the NFC Forum[18], an industry association that promotes the NFC usage and that is composed by hardware, software, banking and telecommunication companies. The NFC is a wireless communication standard defined in the standards NFCIP-1 - ISO18092- and NFCIP-2 - ISO15693 and based on the RFID standards which incorporate the ISO 14443 A and B and FeliCa. This enables interoperability with the already deployed infrastructure of contactless readers and tags following those standards.

The NFC uses the 13.56 MHz frequency and it permits transfers rates up to 424 kilobits per second. The NFC communication is triggered by the proximity between two NFC-compatible devices. This proximity is limited up to 10 cm (Ortiz, 2008), making the NFC more suitable for user-centric applications. Basically what the NFC offers is an easy to use, short-ranged wireless and pervasive interface towards other NFC enabled readers and tags. However, the data exchange through this transparent interface needs a secure element, in which the SIM fits perfectly.

A NFC device can communicate in passive or active mode. In the passive mode it uses the power received from the message of the other NFC device to communicate back, while in the active mode it uses its own power. This leads to three forms of operation:

- Active User/ Reader Mode: When the NFC device is active and communicates with a passive device or tag. For example, when reading an interactive information point, advertisement poster, RFID bar code.
- Passive Tag or Card Emulation Mode: When the NFC behaves as a tag or a passive contactless smart card. In this case, NFC controller handles the radio transfer and the secure element protects the tag information for applications such as payment and ticketing.
- Peer to Peer: When both communicating NFC devices works actively exchanging information through the NFCIP-1 protocol.

This architecture of having the NFC controller connected to the SIM through the SWP protocol enables the SIM to act as the secure element for the radio application, and to offer

[18] http://www.nfc-forum.org/home

more security and portability than by having the secure element residing in the phone. Besides that, the SIM is already globally deployed; it is a standard and interoperable platform for hosting different and independent applications. At last, it can be managed remotely through well defined OTA commands.

The communication between the NFC controller and the SIM is established through commands, APDUs, over the SWP. In order to intercept the tag availability (for the reader mode) or to be read (as in the passive mode), the EVENT_PROACTIVE_HANDLER_AVAILABLE event must be registered in the STK and the SIM applet must catch that event and process the radio specific APDUs.

The communication can also be established between the NFC and the Mobile through a *Midlet* installed in the mobile. The *Midlet* can interface with the NFC Controller through the Contactless API (JSR 257) and with the SIM card or Secure Element through the SATSA (JSR 177). An overview of the NFC architecture in the mobile equipment is presented in the picture below:

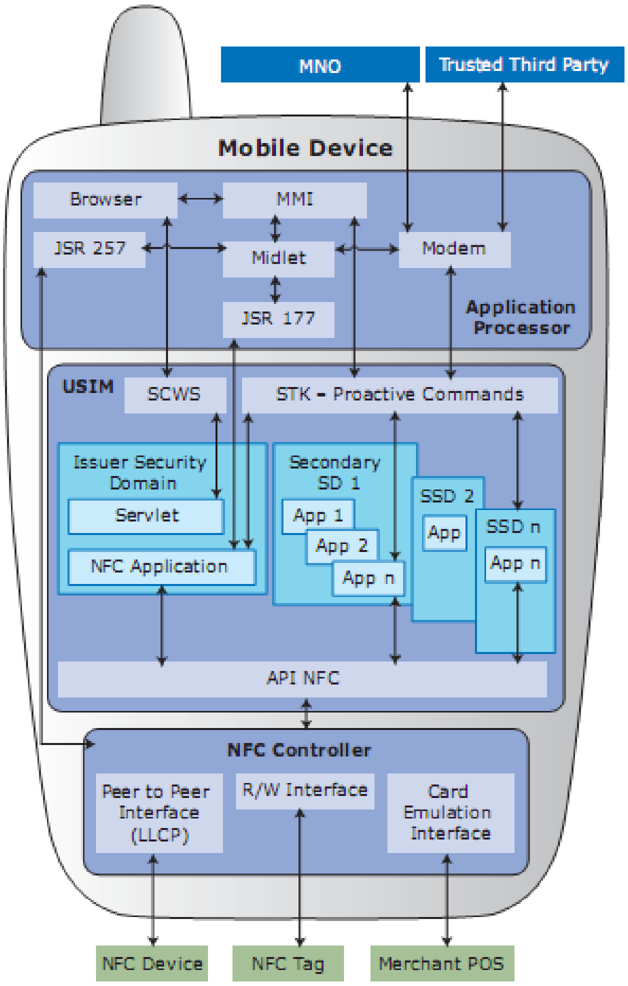


Figure 8: NFC architecture, from (GSM Association, 2007)

Currently the most commented use case for the NFC is mobile payment and ticketing, where the NFC acts as the contactless interface for an e-purse, payment or credit application inside the card; or it acts as the storage of tickets to a service that can be read by a contactless reader. Other application ideas have also been discussed such as physical access, identity management, mobile advertisements and NFC enabled points of information.

In fact, any application is suitable if it can benefit from secure storage and the short-ranged seamless communication provided by the pair NFC and SIM. Moreover, the compatibility of the wireless interface with the standards for RFID and contactless solutions deployed, and the drop of the prices of TAGs and reader, allows the possibility to use these tags to exchange multiple and diverse context information.

This context information can be inherited from the absolute or relative position of tags, but it can also be based on the interactions between the user and the tags (which kind of smart posters he read, which device/reader he has and others). If those preferences are linked and stored with an identity of the user, it may offer a great degree of service personalization.

The availability of handsets implementing the SWP and connecting the NFC to the SIM is still limited. However there are also some implementations of NFC applications by having the secure element in the phone, such as with a few Nokia Models, including the Nokia 6131[19]. In that case, the secure element has a *JavaCard* area for *JavaCard* applications and Mifare area for the security data to be stored.

Currently there are a few NFC trials such as: the German train touch and travel[20], the NFC mobile payment trial on France by Payez Mobile Association[21] and Telia Sonera's solution for delivering public transportation tickets and traffic information[22]. Moreover, a forecast from Frost & Sullivan predicts that "One third of all mobile phones will be NFC-equipped in a span of three to five years"[23].

4.3.2. Application Level Communication

4.3.2.1. IP SIM

A lot of work has been done towards integrating the smart cards services to the internet as it can be seen in (Lu, 2007). As mentioned there, one of the greatest obstacles in the communication between the SIM and the web is the protocol difference between the smart card ISO 7816 and the internet protocols from the data-layer Ethernet protocol until the HTTP application protocol.

One of the most common solutions up to now is to use a middleware that translates the protocol messages. However, building and maintaining a middleware can be quite complex

[19] http://europe.nokia.com/find-products/devices/nokia-6131/technical-specifications
[20] http://touchandtravel.de/site/touchandtravel/de/start.html
[21] http://www.payezmobile.com/media/upload/pdf/140/090210-dossier_presse_uk.pdf
[22] http://www.teliasonera.com/press/pressreleases/item.page?prs.itemId=304418
[23] http://www.ux.uis.no/atc08/workshop/Oehman.pdf

and expensive. Additionally, when introducing another device or application in between the smart card service, you add another point of security failure in the final application.

Due to the recent adoption of the USB as a high-speed interface towards the smart cards by ETSI, full USB cards are starting to be deployed. Those cards will be able to provide better performance and eliminate the need of a middleware for physical protocol conversion. By having the implementation of a native TCP/IP on the card, all middleware needs are eliminated and it remains only the need of getting the internet connectivity from the smart card host. This connectivity can be easily provided through the BIP protocol defined between the SIM Cards and Handsets, or through the USB Ethernet Emulation Model (USB-EEM).

Another importance of the implementation of the USB High speed interface towards the TCP/IP connectivity is that it overcomes the delay introduced by the ISO 7816 for proactive sessions. The TS 102 600, (ETSI, 2009), specifies that the USB UICC may perform a "Resume" signaling to issue a proactive command. Then, the terminal will react within 10-15ms performing the wakeup actions of sending a status message in the ISO 7816 over USB or checking if there is data to be received from the USB UICC in the EEM mode. Through the ISO 7816 physical layer it would be necessary to wait the SIM toolkit polling interval which is at least of 30 seconds (Jurgensen & Guthery, 2002).

The ETSI 102 483 standard defines how the IP connectivity must be established between the terminal and the UICC. In this case, the UICC may play a combination of the following roles:

- A server for a client in the terminal. In this case, the only thing needed is that the terminal must implement naming translation service so the alias "localuicc" can be mapped to the UICC's IP.
- A client for a server in the terminal. Here, there is no need of implementing any routing or address translation. It is only needed that the UICC provides naming resolution so that the applications can target the server IP through the alias "localterminal".
- A client for a server in the network, having the terminal as the IP "gateway". For this case, the UICC must have access to a naming resolution server so it can address the network servers with theirs IP based on theirs alias. It also needs the terminal to do the routing of the packages in between the card and the remote server, and possibly a network address translation (NAT) in case IPv4 is being used.

- A server for a client in the network, having the terminal as the IP "gateway". In this role, the service on the network must be able to resolve the UICC IP address in order to reach it. It is also necessary that the terminal do the routing of the packages in between the card and the remote server, and possibly an address translation, including port forwarding in case IPv4 is being used.

The ETSI standard also defines the internet protocols that must be supported by the UICC and the terminal, such as: IPv4 and IPv6 and its respective ICMP, ARP, TCP, UDP, DHCP, etc... If IPv6 is used, the IETF recommendation is that the UICC uses a stateless address assigned by the network based on a unique value of the UICC, such as its ICCID. In the case of the IPv4, both terminal and UICC will address each other through internal IPv4 address, 192.168.0.1 for the UICC and 192.168.0.2 for the terminal. In the case the NAT is needed, the ports in which the smart card should be accessible are the 3516 for TCP and UDP, and 4116 for both TCP and UDP under TLS.

Furthermore, there is a specification about a SIM application for the IP communication, the IM Services Identity Module (ISIM) which is specified in the standard ETSI TS 131 103. The ISIM application is foreseen to authenticate the device in the IMS IP networks. Actually the ISIM could be used to authenticate in a SIP network even without the whole IMS infrastructure deployed (Gemalto, 2007). Besides being responsible for performing the mutual authentication in the IMS Network, the SIM can generate key materials to establish the IP-SEC tunnel between the handheld and the SIP Proxy.

The Adoption of the IP in the SIM, and as well the ISIM, improves the connectivity of the smart card by allowing it to be reachable and reach other devices through the IP network. It also makes it easier for the SIM to have access to information stored in the World Wide Web, in contrast to the limited content of WAP and SIM browsing domain.

4.3.2.2. Smart Card Web Server

The Smart Card Web Server consists in an HTTP server embedded in the SIM card. It is specified in the OMA Smart Card Web Server Specification[24] whose version 1.1 has just been recently released.

[24] http://www.openmobilealliance.org/technical/release_program/docs/SCWS/V1_1-20090512-A/OMA-TS-Smartcard_Web_Server-V1_1-20090512-A.pdf

The SCWS enables the local web browser running in the handheld to communicate with the SIM, offering a nice, interoperable and dynamic interface in comparison to the STK interface. Besides that, it enables the smart card to communicate in HTML, which is one of the most popular markup languages. The SCWS can target both static pages stored on the card but also applications, as long as they are registered in the SCWS. Those applications can have their dynamic behavior enhanced by an XML processor as described in (Giesecke & Devrient, 2007).

The *JavaCard* applications interact with the SCWS, based on the API described in the ETSI TS 102 588 standard. There are two kinds of SCWS applications: Interception Applications and Content Providing Applications. The first ones are passive applications that intercept request commands targeted to content provider applications but do not answer them. The role of the interception applications is to process the request to generate logs, maintain counters, generate statistics and etc. They only have access to the request URI, HTTP version, HTTP method and header information. The Content Providing Applications have access to the whole message request, including the message body, and they are responsible for providing the return data to the SCWS, once they are the ones really targeted by the request.

The SCWS can be accessed through a BIP channel between the client and the SIM or directly through the TCP/IP stack, if implemented on the card. In the case of the native TCP/IP implementation on the SIM, the SCWS is addressed by the SIM IP and the standard HTTP and HTTPS ports, 80 and 443 respectively. If the connection is done using the mobile phone as the BIP gateway, the localhost IP should be used and the SCWS HTTP server will listen to the TCP port 3516 and the HTTPS Server to the 4116 port.

The SCWS have 2 operating modes: server mode and client mode. The server mode is the one where the SCWS is addressed by the mobile terminal, as the connection to the SCWS from remote applications is not yet supported, and the client mode is used for administration of the SCWS.

The handling of the BIP channels to access the SCWS, in case there is no TCP/IP stack implementation in the card, is handled by the mobile transparently, and the OMA Standard specifies a series of "musts" and "mays" on how they should do it. For example, the terminal must support at least 2 opened channels in "TCP, UICC in server mode" and may open additional BIP channels when an HTTP connection is established in order to support the communication with several applications at the same time.

It is specified in (OMA, 2009) that the SCWS must provide support for HTTP Basic Authentication and may provide support for Digest Authentication to secure access conditions in the SCWS. The SCWS and the SCWS administration agent, actor for the SCWS in client mode, should as well implement HTTPS. For security reasons the TLS keys are stored in a secure area and shall be used by the SCWS or authorized applications only.

An optional security feature that the SCWS may have is an Access Control Policy (ACP) that states which terminal applications should be able to access the SCWS. This requires the terminal to implement an ACP Enforcer, which will retrieve the ACP data from the SCWS (through the HTTP protocol) and then, control the access to the web server. This feature is very useful for the cases when the handset allows unrestricted installation of terminal applications and where a malicious one could try a denial-of-service or other attack towards the SCWS. The figure below shows the ACP Enforcer in the SCWS gateway in the mobile filtering the SCWS access based on the ACP rules.

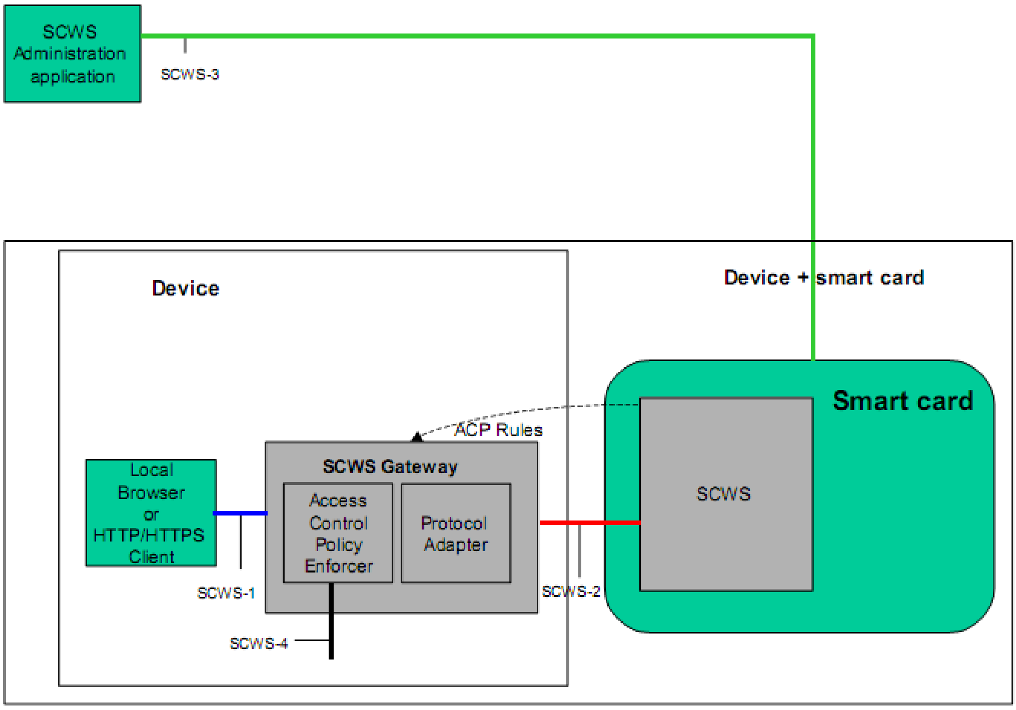


Figure 9: SCWS connectivity architecture, from (OMA, 2008)

The SCWS administration can be done either through the Lightweight Administration Protocol or the Full Administration Protocol. In the light version, the administration commands are sent via SMS encapsulated in envelope SMS PP Data Download targeting the TAR of the SCWS Application. Due to security constraints, those commands must be provided with authentication, integrity protection and sequence numbering in accord to the secure OTA messages. In the other hand, the full protocol is based on having a reliable and efficient transport channel such as the native TCP/IP or TPC/IP over BIP and secured by TLS. The administration commands allow to upload/delete new pages, to change application mapping configurations, to manage users registered for HTTP authentication and to delete data from the SCWS.

The SCWS greatly enhances the final user experience through the friendly and interoperable browser GUI. It also facilitates the application development through the usage of the HTML and TLS, besides creating another channel for accessing the applications stored in the SIM. At last, the Interception SCWS Applications make it possible some recording of context information through the monitoring of statistics about the applications used, and headers of the http requests.

4.3.3. **Java Card** *3.0*

The *JavaCard*, one of the most deployed SIM card platforms, has seen the release of its version 3 just recently. The *JavaCard* 3.0 introduces a great change in the framework, as it defines two editions: the Classic Edition, aimed to resource-constrained cards and basically a smaller evolution towards the previous *JavaCard 2.2.2*; and the Connected Edition, a significantly improved virtual machine aimed to the new high-end smart cards.

The additions to the Classic Edition mainly consist in the support for the SWP protocol for contactless communication and support for the USB interface communication, besides some new cryptographic algorithms that have been added to the API.

The main changes took part in the Connected Edition, aimed to the high-end cards. This edition is based on the Connected Limited Device Configuration (CLDC) J2ME virtual machine which is already deployed in several handsets in the market. It can also be said that the Connected Edition has much in common with the Squawk, the CLDC based Java Virtual Machine deployed for the SUN Spots or small embedded devices (Smith, Cifuentes, & Simon, 2005). The new edition still meets and enhances the requirements of the smart card security.

The full garbage collector implementation, extension of the API (sockets, strings, generic connection framework) to support more classes and features of the Java SE and multithreading are some of the other enhancements to the Connected Edition. It maintains the atomical transactions characteristics of the *JavaCard* by supporting multiple concurrent transactions, nested transactions and the use of annotations. The inter-application communication is also improved by allowing the applications to publish shared services, allowing them to exchange objects ownership and to generate notifications based on events.

The new version maintains the code isolation and applet firewalling features from the previous java versions, but it adds a unified naming scheme for both applications and theirs resources. It also implements complementary access control mechanisms such as role-based access mapped through URIs and security domains. By that, it is also possible to separate sensitive resources of non-sensitive resources.

The Servlet support, multithreading and API support for applet intercommunication enhances the possibilities of having isolated applications in constant cooperation and exchange of data. This allows, for example, an application to handle the treatment of embedded sensors or received data inputs and feeding other applications with the treated data. So, other application could, for example, use the treated data to generate meaningful context information, which could feed high level context to other application.

The target for running the new Connected Edition VM is a device equipped with fast processor (intended for a 32-bit CPU), with more volatile (about 24K of RAM), large amount of persistent memory, full duplex and high speed interface. Network connectivity is also expected as the Connected Edition offers a subset of the Java Servlet API for web applications.

The new *JavaCard* 3.0 fits the high-end SIM card and the new communication interfaces being embedded in it. By that the SIM is transforming it into a powerful networked and embedded device.

4.3.4. M2M SIM

But the SIM's penetration is not only restricted to the mobile market. The SIM cards usage in the M2M (machine to machine) market is in a great expansion. The SIM suits the M2M market as it is a cryptographically secure device which benefits from the GSM connectivity to allow international roaming, network coverage in remote areas and also secure OTA updates. Due to this trend, ETSI is working towards the M2M Standardization[25].

The implementation of M2M solutions are in evidence not only because of their possible automation and cost-saving gains, but also because some regulations are helping this market to emerge. In the vehicle industry for example, the European Commission has launched the eSafety Initiative, which is working towards a vehicle e-call system based on accidents. The European Commission and the industry have agreed on start equipping the new cars with e-calls capabilities in 2009[26]. In Brazil, a regulation voted in 2007 demands that cars produced or imported need to leave the factory with tracking and anti-theft capabilities embedded starting in 2009[27].

However, the M2M business introduces several environment constraints for the deployment of the SIM solutions. In the automotive tracking market for example, the contact surfaces need to robust enough to handle the vibrations of the cars. Vending machines, meters or devices exposed to open air must overcome temperature changes that can vary from a range of -40° C, the winter temperature in some countries, up to around 100° C, for industrial environments. Besides that it needs to be able to handle dust and oxidation.

Another improvement to the current SIM card necessary for the M2M business is raising its life span. The M2M business may requires a lifetime between 5-10 years, besides mechanisms

[25] http://www.etsi.org/website/NewsandEvents/2008_M2MWORKSHOP.aspx
[26] http://www.researchandmarkets.com/reports/295827/wireless_m2m_communication_and_automotive.pdf
[27] http://www.smartcardstrends.com/det_atc.php?idu=8069&PHPSESSID=f5fb2ce6aae01f91240423d0f36859b1

to protect the cards from theft, as the M2M devices may be placed in accessible locations to unauthorized people who may try to tamper it.

Despite those physical conditions obstacles, SIM card vendors such as Gemalto[28] and G&D[29] are already providing special cards for the market. Those cards are being applied in areas such as: fleet management, asset tracking, automation and monitoring[30]. Nevertheless, there are a few points that are on discussion due to the efforts of standardization, such as:

- Limitations of the IMSI range and possible alternative addressing solutions.
- The possibility of changing the operator or deciding the operator after the M2M terminal is bought.
- Lack of security in case the SIM and the M2M device are not coupled with an identity manager.

Those last two points are in great discussion at the 3GPP TR 33.812, (ETSI, 2009), once the coupling of the SIM into the M2M would benefit from the pos-choice of the network operator to be accomplished through a remote personalization (or even installation of the SIM application itself). This imposes a few challenges in the initial connectivity, process of discovery of new operator, and the secure download of the SIM application or personalization.

The 3GPP TR 33.812 also discusses the possibility of having the secure element directly on the M2M terminal, outside of the UICC. But there is some questioning on the security of the terminal as the container of the SIM/USIM application. Moreover, it presents some aspects towards the possibilities of optimization of the signaling used for the stationary devices in order to use the network efficiently.

Anyway, the SIM is to become more and more present in M2M applications, starting from: Alarm Systems, access control, land line backup, automation of tolls, navigation, tracking of vehicles, traffic and fleet management, remote diagnosis, vending machines, information points (smart posters), sensors and devices remote control, metering, stock maintenance, etc.

[28] http://www.gemalto.com/brochures/download/m2m.pdf
[29] http://www.gi-de.com/portal/page?_pageid=44,139339&_dad=portal&_schema=PORTAL
[30] http://www.telenorconnexion.com/business-solutions/

It is not clear if all the mentioned capabilities will be implemented in a single Future SIM, or if we will end up having different specialized SIMs with only a subset of the mentioned sensors or interfaces. In fact, the success of it will face challenges such as being able to set an accessible final price, gathering the support from several different stakeholders (Telecom Operators, Handset Manufacturers, Id Providers, Service Providers, etc) and being able to widespread of the SIM as a development platform.

However, the previous lack of connectivity, user interface and development interface are being countered with the Future SIM. The sum of all the new SIM features turns it into a privileged device which is incredibly small, pervasive, connected and secure. Thus, it is reasonable to expect its adoption.

5. Sensors and Context

The M2M business and the VAS market are pushing the connection of the SIM to different devices, with different capabilities and applications. The mobile phones themselves are incorporating sensors such as accelerometers, light sensors, GPS, proximity sensors[31], etc. This is being pushed towards the SIM card as well. In this section, we will explore a little bit about the power of sensing and context information and how it can be used by the SIM.

5.1. Context Information

We will use the definition of context information given at (Serrano, 2008), where "Context information is any information characterizing the situation of a participant (human or not) in an interaction with its environment."

Context information can answer to questions such as who is the user, who is he together with, where is he, what is he doing and what resources are available for him. There are no de-facto standard categories of context information. This can be justified by the fact that most of the cases exploring context information focus on one or few specific context types and context can vary a lot depending on the situation.

Some examples of context categories are:

- Location: Can be absolute, relative or symbolic location. We reserved a whole section to explain the location context, as it is one of the most used contextual data.
- Environmental infrastructure: Represents the information about which kind of resources are available for the user. For example, if he can roam between different network technologies, or adapt his screen/interface for a more suitable device?
- Physical environment: Includes level of noise, environmental sound, light (presence/absence and which kind of light), vibration/motion, temperature, humidity.
- User activity: Which can be either modeled as something binary such as busy or idle. Or it can be more complex as the representation of what precisely the user is doing, like running, watching TV, or using a certain application in the computer.

[31] http://www.apple.com/iphone/features/accelerometer.html

- Time: Time is a generally easy context to be understood. It can be modeled as absolute time, like 12:04pm in 19/Feb/2009, or relative time such as 5 minutes after a certain event.
- Profile: Personal information about the user or information associated to his identity. It includes data about who he is, what he does, what he likes, who are his friends and which kind of relation does he have towards other specific users. It can also be extended to the emotional context of the user (is he happy, sad, etc) or even to his biological conditions (is he ill, how much glucoses does he have in his blood).

The context information can also be divided in physical context, measured from a physical sensor; and logical context, deduced by monitoring the user activity or information (Hristova, 2008). It can be supplied by sensors or acquired indirectly by accessing stored/shared information about an artifact or person.

One sensor can be used to enhance the awareness of several different contexts. In the same way, the information got by different context information can enhance the certainty about them both or even create the awareness about a different context.

The context information is always to some extent deduced since they carry uncertainties as described in the Heisenberg's uncertainty principle. The information retrieved by the sources has uncertainties due to limitations on sensing devices or ambiguousness or lack of information about the context. For example, if the information that is dark is supplied by a light sensor; it does not necessarily means that it is night, but maybe the device is in a closed environment without light. However, the combination of different pieces of context information can enhance the certainty of the assumption. In the case of the dark sensing, the time information from a clock could help determining if it is day or night.

Due to this uncertain nature of context information, researchers in the area came up with the concept of Quality of Context (Hristova, 2008). Where this Quality of context can be divided in:

- Precision: Which depends on the range of the context measure. In the case of location, for example, it can be the range of measuring of the location sensor, its lowest scale.
- Accuracy: The level of correctness of the acquired data.
- Freshness: How recent is the information.

- Reliability: How available is the information, ratio between the number or times the data was available over the expected amount.
- Granularity: How much the information stands for the analyzed aspect, as an example the temperature in a room may actually vary from the place where the heater is to the door.

The context modeling techniques varies on how the context is parameterized and interpreted. The context data can be modeled as a key-value pair (like Latitude: 40.004761 Longitude: -83.019945) that is later compared in a matching rule. It can also be described as fuzzy variables (Korpipää, Mäntyjärvi, Kela, Keränen, & Malm, 2003) and it can possibly take in consideration the mentioned quality attributes of context. An interesting approach presented by (Schmidt & Laerhoven, 2001) is based on a hierarchical structure, where the data retrieved by the sensors generates several pieces (atoms/cues) of context information associated with it. And then, after evaluating the inputs of every cue, the context is built. A graphical presentation of that architecture is shown in Figure 10.

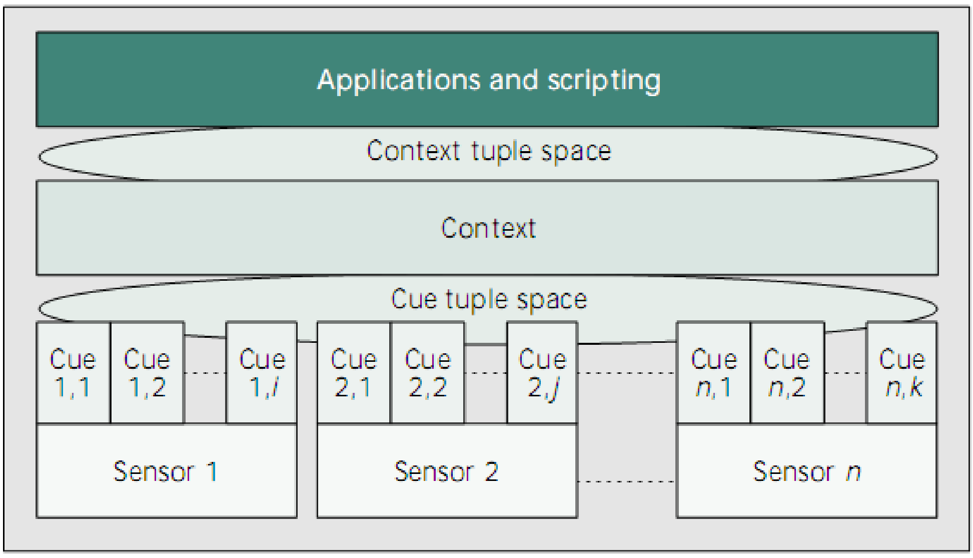


Figure 10: Context-Aware Architecture from (Schmidt & Laerhoven, 2001)

5.2. Sensors

Sensors are basically devices that measure physical parameters and convert them into voltage signals. Those signals are then sampled and converted in binary data to be treated by a computer or microcontroller.

Besides physical sensors there are "virtual sensors" which supplies context information based on applications and software process. The “virtual sensors” can get information from the digital calendar of the user, from his e-mail data, his activity stored data or what application is he running on his computer. At last, there are also the logical sensors, which infers context through some logic based on the information retrieved by both physical and virtual sensors.

Some examples of physical sensors, described in (Meyer & Rakotonirainy, 2003), are:

- Light Sensors: There are a few optical sensors such as photodiode, color sensor, IR and UV-sensors, etc. Those sensors acquire information on light intensity, density, reflection and even type of light (sunlight, incandescent, fluorescent light, etc). This sensing can provide information about the environment as indoor/outdoor location, or if there are light emitting devices like TVs, computers, etc. In general those sensors do not need much processing.

- Cameras: Visual information supplied by cameras can enhance several distinct contexts, but the problem resides on the amount of processing needed to interpret the information from the images or videos. Data about movement and colors require little processing, whereas detection of objects, places, people and gestures require a high processing power. Even though the camera devices are cheap, images require more data storage capacity and they seem more invasive.

- Microphones, audio input: They can provide meaningful information such as noise level, sound frequency or type of sound (noise, silent, music, male/female speaker) using little processing (Beigl, Krohn, Zimmer, & Decker, 2004). Moreover, could be used as basis for generating random numbers or keys, due to its local, random and fast changing nature. But, just like the cameras, they require large data storage capacity and they are invasive.

- Accelerometers and motion sensors: They offer information about motion, inclination and acceleration of the device. Accelerometers are in general implemented with: mercury switches, angular sensors, ball switches. Motion sensors can be implemented by location tracking and IR sensors. A common use of them is to act as a trigger to "wake up" a device. There are researchers and companies work to use those sensors to distinguish user's movements (Karantonis, Narayanan, Mathie, Lovell, & Celler, 2006) or to detect the position of the device in order to choose how to display the information, like with the Iphone.

- Touch: Implemented directly by conductive panes or indirectly through light sensors or temperature sensors. It can presume actions from the user based on what he is touching, or it can detect that an object or place as being "touched".

- Temperature: Can provide either relative temperature as "cold", "warm", "hot" or a temperature measurement. Besides its use in industrial process or product monitoring, it can help to detect if the user or device is in indoor or outdoor environment.

- Humidity, gas sensors and Biosensors (blood pressure, amount of a substance): Those are more relevant for specific applications cases such as aiding firefighters, miners, diabetics, etc.

- Proximity and Location: Can be acquired by relative position to other devices or as an absolute position in a cartographic base. Where the position may come from power measures from wireless radio communication interfaces (Wi-Fi, Zigbee, RFID, NFC), GPS, cellular network location, optical and electromagnetic trackers. Or it can be deduced from the environment. In the IPv4 network, for example, it is possible to infer which country you are by your public IP address. Location is one of the most discussed and possibly most valuable context information. Therefore, we will further discuss it in the next sub-section.

There has been a lot of work on connecting sensors to devices or to wearable objects in order to detect context (Bražinskas, 2008). However, the mobile phone has revealed to be an appealing instrument for collecting context information, due to its pervasiveness. The fact is that more and more sensors are being attached to it.

Many of the new handset models have a wide connectivity (from the regular cellular network to NFC, Wi-Fi and GPS), cameras, motion sensors and even accelerometers. Besides that, as (Bražinskas, 2008) mentions, most of the devices have other sensors that are not open to the applications, such as the microphone, light sensor (used for the camera) and temperature sensor (used for monitoring the battery). This can be justified by the fact that the mobile's first purpose was to be used for mobile communications only. As the usage of mobiles to provide context information and the number of associated VAS tends to expand, those sensors will most likely be accessible through the handset's APIs.

The SIM cards are also getting connected to several sensors. Besides the contactless interface of the NFC which has become a standard, smart card companies are trying to embed new sensors. For example: Oberthur[32] has deployed a SIM card with an embedded accelerometer; Sagem Orga[33] is connecting a GPS directly on the SIM; Telecom Italia[34] has embedded a ZigBee interface in the SIM and Telenor[35] has embed a WLAN 802.11 antenna and radio on it.

5.3. Location Context

We decided to take a closer look at location sensors because it is one of the most explored contexts and there are several methods to capture it with the mobile phone and with the SIM. But before describing the mechanisms to sense the location we introduce a few location-related concepts.

5.3.1. Location Concepts

Physical Location and symbolic location: Physical location is associated with the absolute location of an entity based on a numeric coordinate cartographic system. In the other hand, the symbolic location is context related. A symbolic location can be: at my house, in the living room, at a specific office. A symbolic location can be something more abstract as home environment X foreign environment, or indoors X outdoors.

Absolute location and relative location: As mentioned before an absolute location is related to a numeric coordinate cartographic system such as Universal Transverse Mercator (UTM). The

[32] http://www.engadget.com/2009/02/16/motion-detecting-simsense-sim-card-opens-new-world-of-possibilit/
[33] http://www.cellular-news.com/story/34691.php
[34] http://www.zigbee.org/imwp/download.asp?ContentID=10403
[35] http://portal.etsi.org/docbox/Workshop/2008/2008_06_M2MWORKSHOP/TELENOR_BREDE_WLANSIM%20presentation%20ETSI%20%20june%2008.pdf

relative location is the location towards one or more reference points, such as 20 meters to the right of the shopping mall, above the table, etc. Relative locations can be really useful if the reference context is known. If you know the absolute location of the reference points and the relative location of the interest point, you can calculate the absolute location of the last one.

Accuracy and precision: Accuracy is the closeness between the estimated value of the measure, in this case position, and its true value. Precision corresponds to "the degree to which further measurements or calculations show the same or similar results" (Pichler & Hrachovec, 2008).

Two of the location sensing approaches we will describe, the W-LAN and W-PAN, are based on the interpretation of the position through analysis of the radio signals. Therefore, we will briefly explain how this is done by those radio networks. There a two main approaches that can be used for it.

One of them is to determine merely if the nodes are next to each other. If theirs radio waves can reach other, it means that they are in the range specified by their antennas. This can be useful for relative position and when interfacing with fixed hotspots.

The other approach needs more nodes (three or more for three-dimensional positioning) present, so the location information can be acquired through distance approximation techniques based on lateration and angulation, as shown in Figure 11. As described in (Karl & Willig, 2005), the most important inputs to estimate the distance between each one of the nodes are the Received Signal strength indicator (RSSI) and the time of arrival.

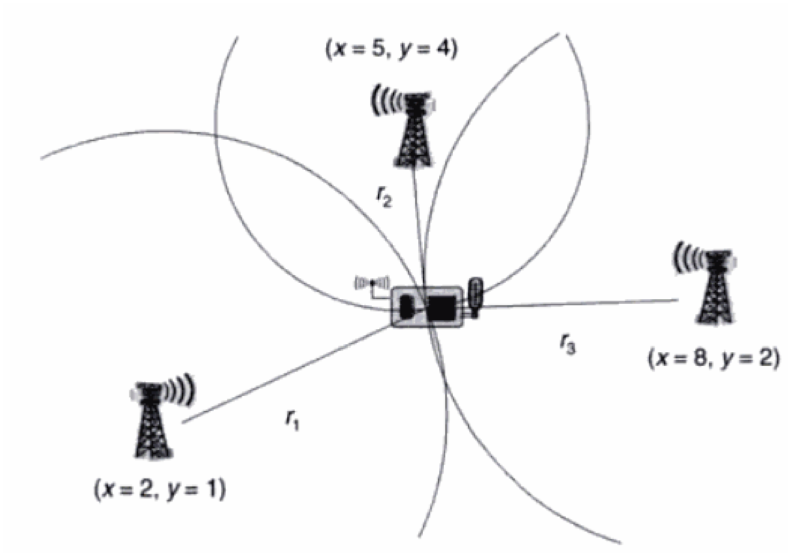


Figure 11: Triangulation by intersection of the signals, from (Karl & Willig, 2005)

The distance can be mathematically calculated if the transmission power, received power and path loss coefficient are known. The strength can also be based in either the theoretical value of the transmitted power, in the measure of it, or in estimations based on packet loss. However, environmental aspects such as node mobility, interference and presence of obstacles can greatly interfere on the estimations (up to errors of around 50%).

The time of arrival can be used as well, but it needs that both the nodes are timely synchronized. Then, based on that, the distance can be calculated on the time elapsed between sending and receiving the packet and the propagation speed.

5.3.1.1. Location Sensors

GPS (Global Positioning System): GPS is a positioning system based on the communication with a network of satellites. As a consequence of that, signal attenuation, bad satellite visibility and multipath phenomenon may limit its usage and accuracy. In general, the GPS works better for outdoor positioning, although it is possible to be tracked inside buildings, with some restrictions. As mentioned before, Sagem Orga is embedding GPS on the SIM and several mobile phones come with embedded GPS.

(Bražinskas, 2008) came to the conclusion that it requires a lot of battery to leave the GPS running as the location sensor without interruptions in a mobile. Therefore, it makes more sense to trigger it once the information is needed.

Cell Identification (CellId): The CellId corresponds to the mobile network cell where the handset is located. It is completely based on the cellular network infrastructure serving the mobile, although the information is shared to both the handset and the SIM card. Since the CellId is needed for the operation of the mobile phones, mechanisms to retrieve it are already implemented on the mobile devices and STK.

However, in order to get the physical location based on the cell location, it is necessary to know the coordinates of the cellular cells. In general just the operators know this information, but as they are somehow static, they could be mapped by a third-party person. Another characteristic of the cells is that they do not have a fixed size. They may vary in radius from 35 km, for the ones located in low density areas such as rural ones, to 100 meters or less, such as the Pico cells that offers coverage in indoor places with high density.

As noticed by the cell sizes, the location information can be somehow imprecise. But a high handover (cell change) rate can also give information about motion. And this can be used to trace user habits or, if matched with time information, to identify the cell corresponding to the person's house, work, etc.

Besides the CellId, the network also broadcast to the mobile the Location Area Identity (LAI) which is based on the country code, network code and an area code. The country code and the network code are publicly know and they can be used for example to detect when someone roam to another country.

WLAN (Wi-Fi): The Wi-Fi hotspots in which the device is connected offer similar position information as the GSM cells in the cellular networks, although its coverage is of a few tens of meters indoors to hundred meters outdoors. It is possible to refine the position measure inside the antenna range with the mentioned triangulation techniques. Wi-Fi hotspots are uniquely identified by the BSSID. But while for the cellular networks the mobile operator has a full map of the cells, the Wi-Fi hotspots have different owners and their position information is not necessarily known by a single entity.

Nevertheless, it is possible to map the hotspots and acquire significant position information and that's what the company Skyhook[36] does. On the one hand, the information may not be so reliable since hotspots owners can deactivate, turn off or move their hotspots; but, on the other hand, the device can detect several the hotspots in the area, so the redundancy in the information can contributes to its reliability.

Moreover, the lack of hotspots can indicate that the user is in a remote area; while a big number of it points that he is in the center of the city. (Bražinskas, 2008) goes even further and points that it is even possible to acquire meaningful information based on the names given to the SSID, as companies and institutions put their names in the SSID.

WPAN (802.15.4): Location has become an important topic within the WPAN. Its importance can be evidenced by the creation of an addendum about the topic in the 802.15.4 IEEE standard[37].

Nevertheless, the range of the 802.15.4 is between 10m in closed environments and 100m in open environments. In order to keep track of distances longer then the 802.15.4a range, the nodes could cooperate and forward their relative distances.

As mentioned before, Telecom Italia has deployed a SIM Card prototype with native Zigbee interface (called Z-SIM), where Zigbee is a high level communication protocol over the IEEE 802.15.4. Some of the characteristics of the IEEE 802.15.4 which favors it to be deployed in the SIM are its low-power consumption and self-configuring capabilities (when powered by a high-level protocol such as Zigbee).

Its capabilities are somehow compared with the RF-ID due to the short range of the communication. However, the IEEE 802.15.4's range is much wider, it is cheaper to implement than RFID readers[38] and the nodes can mutually cooperate. Moreover, 802.15.4 is being widespread in industrial applications, where the M2M SIM is starting to grow, around electric meters and other Business-to-Business (B2B) automation cases.

[36] http://www.skyhookwireless.com/

[37] http://standards.ieee.org/getieee802/download/802.15.4a-2007.pdf

[38] http://portal.etsi.org/docbox/workshop/2008/200812_WIRELESSFACTORY/TELECOMITALIA_Borean_ZIGBEE.pdf

5.4. Sensors and the mobile

The Table 2 presents a small summary of sensors, context information and how present they are in the mobiles and in the SIM.

Table 2: Sensor/Context/Availability Matrix

Sensor	Context	Availability
GPS	Absolute Position (mainly outdoors)	In a few mobiles and being embedded in a SIM
CellId	Absolute Position (with limited precision)	Compatible with all mobiles and SIMs (through the STK)
Wi-Fi	Relative Position and some environment sensing	Present in a few mobiles and being embedded in a SIM
W-PAN (802.15.4)	Proximity, Relative Position, and surrounding environment	Not present in mobiles but being embedded in a SIM
NFC	Proximity Relative Position, and surrounding environment	Present in a few mobiles and integrated with the SIM
Camera (filming)	Environment, motion, relative position	Integrated on mobiles, but the data is too large, it demands much battery and is invasive
Temperature	Environment	Somehow present in the mobiles but not covered on theirs API
Time	Time	Present in the mobiles and accessible by the STK
Light	Environment	Somehow present in the mobiles but not covered on theirs API
Soft Sensing	User activity and profiling	Provided by applications running on the mobile and internet connection
Accelerometer	Motion, device orientation	Present in a few mobiles and being embedded in a SIM
Sound	Environment, user activity and profiling	Embedded in handsets, but may be invasive and require a lot of storage.

Thanks to the abundance of sensors connected directly or indirectly to the Future SIM, it can play a role of a complex context-aware platform. And as those sensors start to be integrated directly in the SIM, the security of the sensing data is enhanced, resulting in a trustworthy source of sensing evidences.

Moreover, the presence of so many location context inputs makes it possible the development of a very efficient high lever location sensor around the SIM. By default it could always acquire location information from the CellId, and when suitable it could activate other sensors (such as the GPS, Wi-Fi positioning, etc) to enhance the precision of the location. This suitability could be based on:

- The available power, as some sensors consumes more energy.
- The surrounding environment, as some mechanisms work better in indoor/outdoor environment.
- Preference for relative or absolute location data.

6. Identities

As one of the most important usage of smart cards is in the identity domain and those also represent a pointer to several attribute linked data, we will review some the smart card and SIM card towards Identity Management. We will take a closer look at some National Identity standards since they are widespread and a large number of them are migrating to the smart cards. But we will also assess some industry identity standards as they explore other characteristics of the identities, such as attribute providing.

6.1. Identities Fundamentals

One of the common mistakes is to confuse identification with authentication. As explained in (Riley, 2006), identification corresponds to an assertion about "who" (or "what") is the user or system. One important characteristic of the identity is that it is public, once it corresponds to the claim itself but not to the fact that it is true or not. It can be represented by IDs, username, digital certificates, ATM cards, etc.

In the other hand the authentication corresponds to a method of proving that that identity belongs to you. Differently it is not public, and the possession of this secret is what proofs the identity claim. Authentication can be based in what the person knows (such as passwords), what the person has (such as a personal device) and what the person is (such as his biometric information). Whereas, when two or more of those are used together to authenticate the user, it is considered a strong authentication case.

At last, authorization means giving permissions (access control, both physical access or data access) to the authenticated user. The authorization can be materialized through a token or ticket, as this token gives the user privileged rights. The differences between the three concepts can be summarized in the Table 3.

Table 3: Identity X Authentication X Authorization adapted from (Riley, 2006)

	Provided By	**Answers**	**Attributes**
Identity	Id Provider	“Who are you?”	Public assertion
Authentication	Id Provider	“Ok, how can you prove it?”	Secret Response
Authorization	System	“What are my privileges?”	Token or ticket providing access

Once we say that the Identity is a claim, we will present as well the definition for a claim. In order to do so, we will use the definition given at (OASIS Web Service Secure Exchange TC , 2007), which represents a consensus between the several organizations behind OASIS[39]. Their definition is: "A claim is a statement made about a client, service or other resource (e.g. name, identity, key, group, privilege, capability, etc.)".

For the definition of digital identity we will use the one from (Cameron, 2005), which is one of the most referenced texts in the identity area. There, digital identity is "a set of claims made by one digital subject about itself or another digital subject"; where a digital subject is "a person or thing represented or existing in the digital realm which is being described or dealt with". In those conditions, trust is an attribute connected to relationship between the subject that set the claim, the claim’s subject and the ones assessing the claim.

Whenever you are using the internet, or actually any machine interaction without human supervision, your identity is disclosed until a mechanism force you to identify yourself. As the famous cartoon from Peter Steiner say: "On the Internet, nobody knows you are a dog". The only information you have a priori about an internet user is its IP address, which in this case it is kind of identity, but not enough to say that there is a user behind the device.

This lack of a high level identification has lead to the creation of several mechanisms to identify a user towards a service provider in the virtual world, and the most common one is through the creation of a user account. This approach is quite natural and similar to what happens in the outside world. There, if you want some range of services (for example a newspaper subscription, a bank account, to be an associate in the gym), an identity is created

[39] http://www.oasis-open.org/home/index.php

for you. This identity can be represented by a plastic card, with or without physical identification features, or just by an entry in a database.
The user account is generally linked to some information from him. This information could be a physical address, credit card information, or merely an alias for a forum. At least one of them, traditionally the “user name”, will represent the user’s identity. Meanwhile, the other data can be gathered for Customer Relationship Management (CRM) purposes or security concerns. The secret information used for proving the identity in the web has traditionally been modeled as a password, despite other mechanisms, such as physical tokens or biometric information, can be used.

As just mentioned, establishing a digital identity towards a service provider may rely on sharing some personal information, in order to strength the identity claim made towards him. (Cameron, 2005) discusses that the more information you store about a user, the higher the losses when suffering a breach. Thus, in order to have the customer consent to give his information, it is necessary for the user to have a trust relation towards the identity provider and on how is he going to use his information.

But in other hand, the more information a service provider has, the more accurate and trustful profile he can build of its users; and, therefore, he can offer more relevant services to the user and adapt them to his needs. But since the profile is linked with the user, and with the SP, a business differentiation emerge dividing a service provider role that would focus on the service and an identity provider/profiling role capable of storing securely the user data, and managing the sharing of his information in accord to his control and acceptance.

The relationship between the Identity Provider and the identified user is what shapes the privacy of the user. The identity provider needs information about the user to provide him his identity. And the user must agree with the provider towards the scope of the usage of that data.

By having this identity profile centralized, it would be possible to personalize the user services without having him to give his information every time he carries business with a different service provider. He would only need to define the policies for the access of his information. And, the new service provider would just need to trust the identity provider.

Those topics are in fact related with Federation and Attribute Services and will be discussed later on in this chapter.

Anyway, the lack of a single common standard in the industry related to identities, the numerous legacy systems where the users already have accounts and the fact that users are getting used to create accounts for each systems; have evolved to a point that the user has to manage several digital identities and several passwords (generating a phenomenon called password fatigue). This leads to bad management of passwords, such as writing passwords or reusing the passwords[40].

Another complication in the world of the digital identities is linked to the identity of the service provider. The absence of a common and well-understood (by the user) framework for the identification of the service provider results in "phishing" activities, which leads the user to disclose private information to illegitimate parties. Often users do not pay attention if a mutual authentication procedure is being carried or if the web site credentials are true or not (Cameron, 2005).

In summary, possessing many different and unrelated identities can lead to poor management of those, such as:

- Usage of weak authentication mechanisms that allow the identities to be stolen; keeping track of several accounts and passwords leads the user to follow insecure practices such as re-using passwords or using simple passwords[41].
- Not intentional disclosure of data, by giving information without knowing how it is going to be use.
- Misuse of disclosed information. The service provider, or someone working there, can misuse the information stored on their databases. The banalization of the agreement terms signed on the internet and service level agreements (SLAs) ends up not protecting the user identity.

Nevertheless, those issues are leading to the development of standards and solutions for having a single storage of information or a common protocol for interoperability between the multiple systems where the users have identities registered.

[40] http://www.readwriteweb.com/archives/majority_use_same_password.php
[41] http://www.darkreading.com/blog/archives/2009/02/phpbb_password.html

In the corporative area, LDAP[42] and Active Directory[43] have already been used to provide a "single" directory service that feeds all the systems in the company with the identity data of the registered users. They make it unnecessary to provision all the different systems with data and to update and manage the data only at the directory service.

However, the problem is when there is the need of interoperating those directories with the external network, once the different directory systems do not necessarily understand each other. Single-sing-on (SSO) services and Federation correspond to a solution for that problem.

The SSO stands for the usage of a centralized directory server which authenticates the user, then, act as the entity that will authenticate him in another system, avoiding the user to repetitively enter his credentials. The federation concept stands for enabling the identification services to be ported and accepted in between different security domains. The federation is what makes it possible the cross domain SSO, information sharing and dynamic account provisioning. Federation represents the agreements between entities, often implemented through cryptographic mechanisms and backed up by SLAs, to cross domains and enable seamless business interactions by allowing the user to authenticate himself locally and access resources globally. (OASIS, 2005)

In general, the components of an identity framework are: the subject, the Relying Part (in some cases represented by the Service Provider) and the Identity Provider (IdP). The subject is the one that presents an identity, make a claim. He does that in order to acquire certain rights or privileges towards the Service Provider (SP). The SP, in order to offer the rights to the subject, needs to authenticate the subject's claim. In order to do so, it needs the subject to authenticate with the IdP, who will issue a token, a signed statement, confirming that that identity really belongs to the user (based on the authentication). Thanks to the trust relationship between the SP and the IdP, and IdP and the user, the SP accepts the token and gives the rights to the user.

[42] http://tools.ietf.org/html/rfc4511

[43] http://www.microsoft.com/windowsserver2008/en/us/active-directory.aspx

6.2. National E-IDs

Countries usually issue several identities to its citizens, such as passports, driver’s licenses, national identities, tax card, etc. The passports, as international documents, comply with international specifications ruled by the International Civil Aviation Organization (ICAO) and there is less flexibility in their implementation. Meanwhile, the other documents standards are generally ruled by national governments and can be quite different from one country to the other.

Nevertheless, the general of the reasons for implementing a national E-id card, based on (Arora, 2008) and (Slagmolen & Pastors, 2000), are:

- Support for e-government services, either to enhance the citizen participation, to cut costs, or easily gather data from the citizen.
- Enabling identification, authentication and signing capabilities. One of the drivers to push this a goal is the European Directive on Electronic Signatures that has been issued in 1999.
- Prevent fraud by making the identity card harder to counterfeit.

A representative list of e-government applications deployed in E-id cards can be found in (Arora, 2008). Some of them are: age verification, for cigarette vending machines for example; checking personal data; purchase of tickets for public transport; e-voting; managing information change, such as address change; authenticate and digitally sign online tax submissions; and send authenticated and encrypted emails. Unfortunately, as the same analysis report mention, there is a big lack of information concerning the usage of those e-government applications. In general, there are only public statistics on the number of issued cards, but not on the use of the mentioned services with the cards.

While some countries have taken the approach of offering a single format or a single valid ID card, others allow multiple instances of the same identity such as the case of the Austrian Card. Or, for example the FINEID, which enables the user to have his certificate in his citizen certificate in his E-ID card, Bank card or SIM card.

In what stands for the identification of the user, the card clearly address the identification factor represented by "what the user has". But, it can also be enhanced or extended to provide authentication using other identification factors such as biometric data ("what the user is") or a password or PIN Code ("what the user knows").

6.2.1. FINEID (Finnish Electronic Identity)

The FINEID (Finnish Electronic Identity) user certificates, which are from the X.509v.3 type, are stored in the smart card only and there are no copies of them in a government database. This was the schema chosen by the Finish government in order to provide more privacy to the user. The smart cards contain one certificate for authentication and encryption and another one for non-repudiation digital signatures, compliant with the European Act on Electronic Signatures.

The FINEID application can be deployed in the FINEID Identity Card, in Bank Cards or in SIM Cards (that were available from Telia Sonera[44] and Elisa operators). In the last case, the private keys are personalized in the card and deleted from the production data afterwards, while in the first two cases the keys are generated by the application inside the card. In order to perform an operation with the keys, the user must enter a PIN code (Finish Population Register Centre (VRK) , 2005).

In the SIM card case, the key pairs are generated during the card personalization, and PIN needed to use the keys is sent to the subscribed in a sealed envelope. The customer must also register the mobile citizen certificate towards the certification authority. The description of the registration procedure and the information necessary to the user to check the integrity of the card are sent to him by the issuer as well (Finish Population Register Centre (VRK), 2004).

The SIM Card in usage is a SIM/USIM with a RSA cryptoprocessor and storage space for the key pairs. The card also contains the SAT microbrowser WIB, including the WIB plug-in used to generate signed SMSs. This allows the SIM to be used as another identification factor to a web transaction by adding the mobile user's signature to a transaction or to authenticate and protect the integrity of the SMS for a SMS based service. The FINEID application file system is based on ISO/IEC 7816-15. The supported commands derive from both the ISO 7816-4 and ISO 7816-8.

[44] http://www.epractice.eu/en/news/284000

In the case of the FINEID, the government issues the certificates not only to the final users but also to service providers. In this case the service providers are responsible for keeping their private keys secured. The government is also responsible for maintaining the public key's database as well as the Certificate Revocation Lists database. Thus, before an operation the public key can be retrieved and the certificate validated against the revocation list. The government, represented by the Finish Population Register Centre (VRK), acts as the Certification Authority (CA) and is the responsible for signing and validating the services and citizens certificates.

6.2.2. Austrian Citizen Card

The Austrian Citizen Card deploys an identity management system where the user ends up having a distinct identity towards each government sector, preserving his privacy. In order to do so, he Source PIN Registration Authority, represented by an organ of the Austrian Government, is responsible for creating a source PIN (sPIN) for the citizen based on the diversification of his PIN. The PIN is a unique personal identification number which belongs to the user, and, for privacy reasons, is not used to identify him. The Source PIN Registration Authority keep the secret key and the nonce used in the sPIN derivation, but for privacy reasons it does not keep the sPIN. This last one remains in the sole control of the user. The authority also creates an identity-link between the user's sPIN and his public key, and it signs that identity-link. (Gert, 2007)

Then, for further protecting the user's identity, he uses a different ssPIN (sectoral specific PIN) for each different institution he is dealing with. The ssPIN is generated through a hash function from both the sPIN and a specific alphanumerical code from the governmental sector to whose application he is being identified. In that way, the user is identified just through his single ssPIN, but his sPIN or PIN can't be retrieved. Thus, it is not possible to perform cross associations between the services used by the user, preserving his privacy.

Besides the mentioned privacy concerns, the Austrian Identity differs from the others by being possible to host it in several different platforms. The definition of the citizen card security layer is abstract enough to enable its implementation not only in smart cards, but also directly on the mobile phone, or other devices (Roessler, Posch, & Hayat, 2005).

The Austrian case makes it possible to interoperate with other E-Id schemes that follow the European Signature Directive. For the foreign E-Ids, a recurring identity is created to substitute the sPIN, based on a unique value of theirs E-Id. The Source PIN Register Authority needs to create the Identity-link based on this Substitute Source PIN (the foreign identity) and a public key signature as well. After that process, the Identity-link can be used for e-government application in Austria just as the Identity-link created from a sPIN of an Austrian citizen.

6.2.3. MyKad: The Malaysian E-Id

One thing that makes the Malaysian E-Id card (the MyKad) quite special is that it really makes use of its multi-application power. It carries the biometric information corresponding to thumbprint image and colored photo. Where, the fingerprint is encrypted in the card and compared with the actual thumbs placed on a scanner, providing an Automated Fingerprint Identification System (AFIS).

By default, only the National Identity card application is loaded into MyKad, but other applications have been developed and may be loaded voluntarily. Between the available applications there are passport information; health information (portable records of basic medical data accessible only to authorized medical personnel); Touch n Go application for toll, public transportation, parking and cashless payments; PKI signature: bank ATM application and loyalty application. It is an identity card that contains both government and private sector applications. Furthermore, the applications can be possible to be personalized after the MyKad has been issued[45].

According to (Unisys, 2008), it has been issued MyKads to more than 18 million citizens, which makes it one of the most spread E-Id cases. However, (Looa, Yeowa, & Chongb, 2009) points that one of the reasons for the widespread of the card in between the population is based on the cultural characteristics of the Malaysians on accepting the decisions made by the government without much hesitation. The article argues that the Malaysians tend to not worry much about privacy in comparison with other countries.

[45] http://www.malaysiacentral.com/article_people_of_malaysia/mykad_the_government_multipurpose_card_frequently_asked_questions.php

6.3. Industry Standards

Besides actions from governmental and local authorities towards identity management, big players in the IT industry have been working towards the creation of standards for managing digital identities, securing them and providing interoperability. We will describe the WS-* and SAML as they are the common building blocks for IdM solutions and the two IdM frameworks: Microsoft Cardspace and Higgins.

6.3.1. WS-* and SAML

WS-*[46] (or WS-Star) corresponds to the Web Services specification under the WS-Security standards defined by OASIS. The Web Services themselves already provide interoperability between different softwares, as HTTP can be used as a standard communication channel between those different systems, SOAP (Simple Object Access Protocol) as the standard message format containing the XML data objects, and WSDL (Web Services Description Language) as the descriptor of the Web Service.

A good representation of the whole WS-* stack where the Web Services are defined can be seen in the Figure 12.

[46] http://www.networkworld.com/news/2005/071405-ws.html

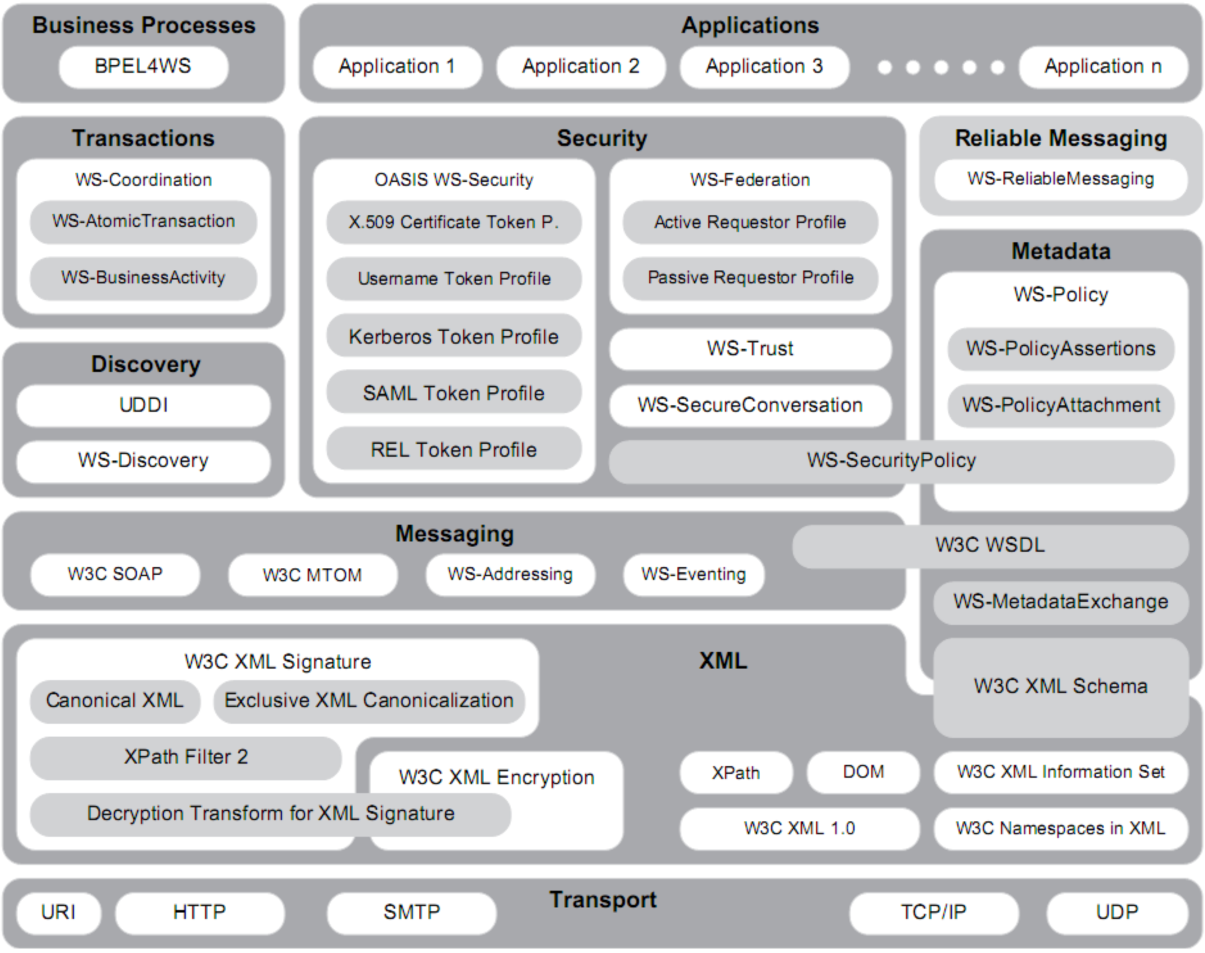


Figure 12: WS-* stack from (Geuer-Pollmann & Claessens, 2005)

Since our goal is to discuss the identity related standards, we will skip the discussion over the standards that are not that much related with identities (such as WS-Eventing, WS-Transactions, etc). We will focus on briefly describing the WS-Addressing, WS-Security, WS-Trust, WS-Policy and WS-Federation as they are the building blocks for several industry standards.

The WS-Addressing provides a general method for addressing Web Services, independently if they are deployed over the HTTP protocol. Besides adding information for the localization of the Web Services endpoints and possible intermediary nodes, the WS-Addressing can be used to assign a unique ID to the message, the "MessageID". Then, if the “MessageID” request a reply message, this reply must reference the “MessageID”. The uniqueness of the “MessageID” works as a mechanism to counter replay attacks.

The WS-ReliableMessaging describes a protocol of acknowledgment and retransmission based on the inclusion of unique sequence number identifier in the SOAP message sequence. This ensures reliability to the messaging, and, if used in conjunction with the WS-Security, WS-Secure-Conversation and WS-Trust, it can protect against attacks at the network layer.

WS-Policy defines capabilities, requirements and constraints policies on the web server, such as stating that the requests must be signed with a particular key type, or followed by a particular token or other. Some of the policies that fit into the category of security policies, such as the token requirement, are in fact defined in the WS-SecurityPolicy (Geuer-Pollmann & Claessens, 2005). However, it is the WS-PolicyAttachment specifications that define how to attach the policies to the Web Services and it is the WS-MetadataExchange which defines how to retrieve the policies (or other metadata) associated to a web-server.

WS-Trust is the specification that describes the "security token services", which are responsible for generating, validating and renewing the security tokens corresponding to the SOAP messages. It even allows a Web Service to act as inter-domain broker and provide the service of converting tokens from one format to the other. It introduces elements such as timestamps, expiration time and challenge-response extensions to the Web Services, in order to counter man-in-the-middle attacks. Together with the WS-Policy and WS-Security, it is the building block for an implementation of an IdM framework over WS-*.

The WS-Security, or SOAP message security, provides an extension over the SOAP, through the addition of security headers, in order to ensure confidentiality, integrity and data origin protection to the SOAP messages. The standard describes how to apply the XML Signature and XML Encryption functions to the SOAP message parts. It is also possible to encapsulate several security headers as long as each security header is targeted to a different SOAP intermediary. Thanks to a few WS-Security extensions, the token formats supported includes: username token, X.509 Certificates, Kerberos tickets and SAML tokens. The WS-SecurityConversation specifies the possibility of establishing a security context between the endpoints based on session keys.

At last, the WS-Federation defines mechanisms to allow, through WS-Security, WS-SecurityPolicy and WS-Trust, the exchange of security information between different domains. It enables requesters from one domain to get a security token from a different domain, as long as both domains maintain a federated relationship. The WS-Federation also specifies a pseudonyms service so that the privacy of the requester can be assured.

The SAML (Security Assertion Markup Language), in the other hand, is a XML-based framework that defines assertion messages, protocols for assertions exchange and bindings of those protocols onto standard communication protocols (OASIS, 2006).

SAML is designed to be used by other standards and it has been adopted by both Liberty Alliance and by the OASIS Web Services Security (WS-Security) specifications. Where, in the WS-Security, SAML is mainly used as an assertion security token.

The SAML Assertions encompasses:

- Authentication statements which describes the mechanisms and time stamp used to authenticate the subject.
- Attributes statements which contains attribute assertions about the subject.
- Authorization statements which defines the grants of the subject.

Through the mentioned assertions SAML works as a security token for SSO, authentication requests, assertion queries, name identifiers and mapping.

6.3.2. Cardspace (and Geneva)

The Windows Cardspace[47] is a framework from Microsoft which allows the management of identities by presenting the identity information to the user in a user-friendly manner, as a portfolio of digital cards. Cardspace has been implemented in the Windows O.S., but the specifications of the used identity metasystem are open and built over the WS-* protocols, so any one can implement it and interoperate with it (Veugelen & Gilis, 2007).

The identities are represented by Information Cards, in a digitally signed XML format, which are stored in the user's computer. The identities only represent the relationship between the user and the respective IdP. A corresponding security token will just be created when the user wants to prove his identity and asks the IdP to authenticate his claim. An illustration of this

[47] http://www.microsoft.com/windows/products/winfamily/cardspace/default.mspx

token creation can be seen at .The security token formats can vary from Kerberos Tickets, SAML tokens, XrML and X.509 certificates.

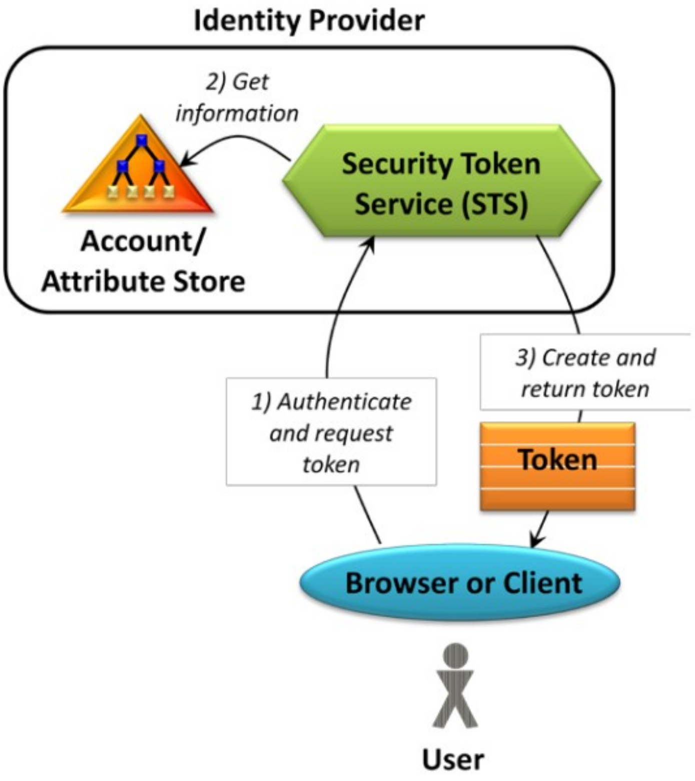

Identity Provider
2) Get information
Security Token Service (STS)
Account/ Attribute Store
3) Create and return token
1) Authenticate and request token
Token
Browser or Client
User

Figure 13: Process of token creation, from (Chappell, 2008)

Cardspace also lets the user create self-issued cards to be used to fill registration and login forms in websites, and, later reuse those cards on multiple websites. This tackles the password fatigue problem cited previously. It also embeds more security to the process, as the authentication is done through SAML tokens which can not be easily stolen and reused, due to public key signatures, time stamping and validity periods. Moreover, it improves the user experience as he does not need to re- provision his data.

The cards that are created and managed by the IdP can be used to provide attribute assertions about the subject to which the card is related with. In theory both self-signed or managed cards can provide information about the user, but in some contexts only the managed cards information may be useful due to the trust position of the IdP.

Another particularity of Cardspace is that it forces mutual authentication so the online service must also identify itself towards the user, and, consequently, prevent phishing attacks. The selection of the identities is done by an identity selector software and the access to those can be protected by a password. The identity selector is presented by Microsoft as a trusted tool to avoid phishing, besides being an easy interface for the effective management of identities by the user. The selector merely provides the selection of the identity. The authentication method towards the IdP is still up to the provider requirements.

Recently, Microsoft has announced an evolution of the Cardspace project known as Geneva project[48]. The Geneva project consists on the Geneva Server, Geneva Framework and Cardspace Geneva. It is completely based in the WS-* open standards and on the concept of the claims, where the relying part states which claims it needs and which IdP it trusts, and then, the user ask the IdP for a security token that provide those claims.

By implementing all Geneva Server, Geneva Framework and Cardspace Geneva, Microsoft trails a step towards offering a framework for the whole IdM chain. The Cardspace Geneva corresponds to the User Interface towards the IdPs. The Geneva Framework provides the API building block for the application to be able to request, verify, access claims, etc. Finally the Geneva Server corresponds to the IdP System, that acts as the Security token Service (STS), and which is actually built over the Geneva Framework. Nevertheless, all those components are built over the WS-* standards and they can interoperate or be replaced by other WS-* equivalent implementation from any vendor and still offer the same IdM capabilities (Brown & Mani, 2008). The Figure 14 positions the three Geneva components in the IdM model and presents the process for the user to submit the token containing the necessary claims for the SP application.

[48] http://www.microsoft.com/forefront/geneva/en/us/

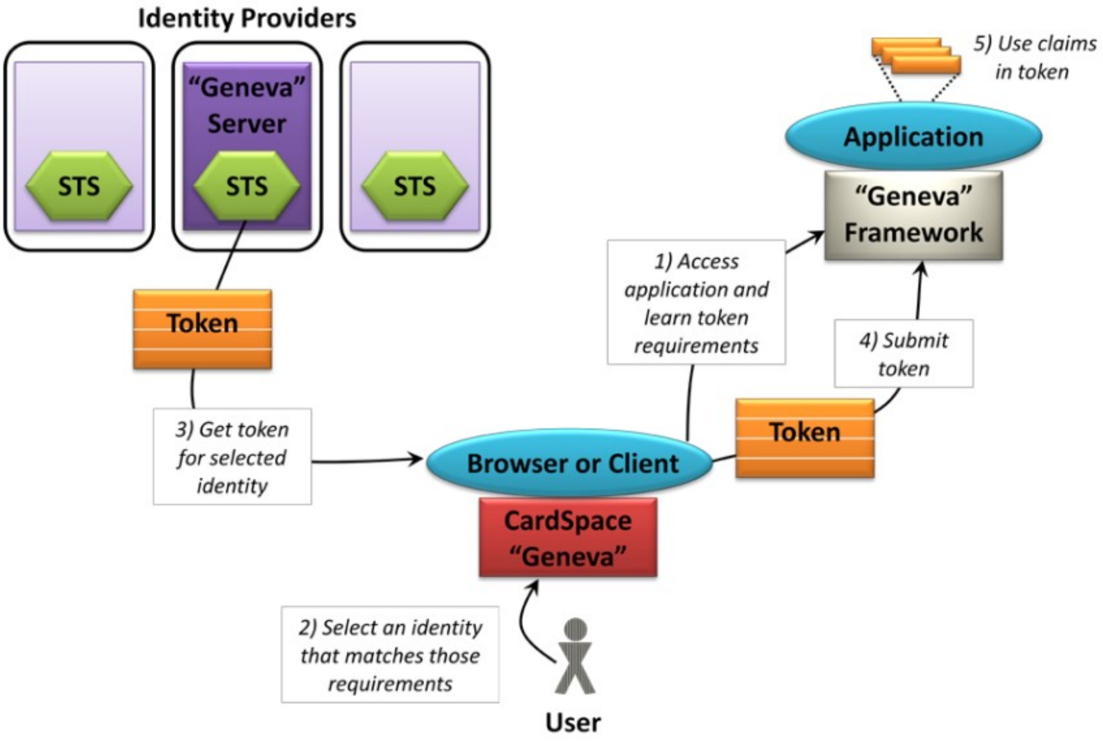


Figure 14: User providing his claims to SP through Geneva, from (Chappell, 2008)

The Geneva Framework support claim delegation as well, once it is somehow equivalent to a claim format transformation. But, in this case, the IdP receives a security token from an application; and, then, based on that token, it issues another token for the claim statement towards another application. This maintains the user privacy, since just the security token has been forward, but not his authentication credentials such as username and password.

In what concerns the Identity Selector, the Cardspace Selector allows an option to export the identity card features, allowing them to be used in another environment than the one where they were installed. The exported identities are protected with pass-phrase selected during the exportation process.

6.3.3. *Higgins Project*

The Higgins Project[49] does not attempt to create a digital identity protocol, but to integrate digital identity, profile and social relationship data across multiple stakeholders and through existing identity protocols and frameworks such as Microsoft CardSpace, WS-Trust,

[49] http://www.eclipse.org/higgins/

OpenID[50], SAML, XDI[51], LDAP and other. The project is composed by 3 areas, the Higgins Client, Identity Web Services, and the Higgins Identity Attribute Service.

The user has an identity selector, which allows him to choose which one of his identity elements, represented through i-cards, he wants to share. The i-cards encompass both cards that are issued by the user and which claims information about himself, cards that are issued by IdP making claims about the user to Third Parties and Relationship cards that allow entities to share claims. Similarly to the CardSpace Identity Selector, the goals are around offering the user a friendly way to manage his identities and allowing him to have a "universal login" tool. The Higgins project also mentions that in between their objectives it is to facilitate the data profile and social relationship information exchange between the different identities of the user. It is worthy to mention, that the Higgins selector has a switch that can invoke the proper Higgins I-card selector, or Microsoft CardSpace selector or other, allowing different card types to be used within the application.

The Identity Web Services offers libraries for STS (Security Token Service) IdP based on WS-Trust and on SAML2 assertions. As well, it provides java code to be used by the relaying parties, through what it calls Extensible Protocol RP. This protocol supports authentication using the Information Card and methods for verifying the tokens. It plays a role similar to the Geneva Framework mentioned in the newest CardSpace version.

The Higgins Project clearly defines the Identity Attribute Service (IdAS) which is built primarily to provide context information about the user conforming to a context data model whose sources are the ones the user share relations such as enterprise directories, social networks, RDF repositories. The attribute context data follows the RDF & OWL semantics. The IdAS provide a Java API that the context providers may use to share user data and to convert the information datasets into a RDF/OWL-based representation called the Higgins Context Data Model (CDM).

[50] http://openid.net/
[51] http://www.xdi.org/

6.4. Biometrics

Biometrics corresponds to the recognition of an individual based on the measurement and analysis of his physical and behavioral aspects. Some biometric techniques include: fingerprint, iris scan, face recognition, DNA, hand geometry, voice recognition and hand-writing patterns. The biometric information can enhance identity, verification and authentication mechanisms as it consists into a unique feature that can identify a user.

In fact biometrics is massively deployed in several physical identity cards that carry a picture of the owner. This picture, a facial biometric, is aimed to present something that can be used for a visual verification on the side of the agent (Arora, 2008).

One of the biggest concerns about the usage of biometric information is on having the biometric template, the synthesis of the biometric characteristics, stolen. Since the biometric template can not be revoked (a user can not revoke and have its physic characteristic such as fingerprint or iris reissued) this is a very important topic to be taken into account. A solution for that is the storage of the fingerprint information in a secure environment, such as a smart card. It enables the possibility of employing match-on-card (MOC) identification without the need of transmitting the biometric information outside of the card. Inside the card, the biometric information can serve as one authentication factor complementing or replacing passwords. Despite MOC solutions on the regular smart cards, as the example of the Portuguese E-Id (Card Technology Today, 2008), there are already deployments on the SIM[52].

As shown in (Card Technology Today, 2008), the memory needed to store the biometric information is not so much, especially if you take into account the new high-density smart cards. A facial image can require 20KB while the iris image can require 30KB and a fingerprint 8KB. If instead of using the image, the biometric template is used, the size requirements are reduced by around 90% or less.

[52] http://www.precisebiometrics.com/filearchive/3/3662/Match-on-SIM_LR.pdf

What security experts such as Bruce Schneier[53] and (Riley, 2006) discuss is that biometrics should not be used as an authentication secret, but as identity information. By that, the identity, biometric data publicly known, identifies the user, but in order to obtain authorization in a system, a secret is used. This argument is based on the fact that biometrics can be tampered: they can be scanned; they are left when people touch objects; and people can be filmed without their consent. Moreover, differently from the secret, the biometric can not be revoked.

The MOC solution in the smart card presented before has the biometric in a context that is hard to characterize between identity and authorization secret. It is something in between, since the biometric information is actually the input to authenticate the person which has the card, but the biometric alone is powerless and the card can be revoked. The fact that the biometric template is not distributed and it is secured in the smart card makes the MOC a secure use of biometrics.

6.5. SIM Card and Identities

As mentioned earlier in this chapter, many countries are trying to strength the security capabilities of their national IDs, passports and relevant documents. Those documents were usually protected against forgery and misuse mainly through special techniques used in the card body design and its manufacturing material, such as watermarking and UV printing (Liersch, 2008).

The introduction of the smart cards allow to store much more data since the data space becomes limited by the memory size of the chip instead of the physical size of the card body. Besides that, smart cards can optimize the verification process of the data, allowing M2M interfacing. It also enables to store other applications and it is a tamper-proof device.

International Traveling security requirement have also pushed to the introduction of E-passports. Where the biometric data of the owner is stored in machine readable format and made available to the readers through a short range wireless interface. The data should be protected through a Public Key Infrastructure and secured in a SE such as the smart card.

[53] http://www.schneier.com/blog/archives/2009/01/biometrics.html

"More than 65 countries are now issuing one form or another of electronic passports, with numerous more in the pipeline for 2009 and beyond." (Elsevier Ltd., 2009)

Furthermore, identities are already stored in smart cards for other purposes such as hotel keys, employee identification inside the company, student identification. The multi-application aspect of the smart card allows the same card to be used in different scenarios. For example, a smart card student card can be used to identify him, serve as a library card, a wallet for the cantina or the student village laundry.

The SIM Cards by default are already a user identity as it identifies him in the telecom operator network. Actually it corresponds to a quite complex case of identity management, once:

- There is some profiling as the user often needs to provide some personal information during purchase or activation of the SIM card or mobile subscription.
- It handles the unique identifiers MSISDN and ICCID, which are provisioned in telecom platforms and used to target individuals.
- There is an authentication infrastructure that confirms that the MSISDN corresponds to the SIM Card by performing the GSM authentication based on the keys securely stored in the cards and in the Operator's database.
- They already implement a successful example of federation which is the roaming agreements in the telecom field. They allow a user to log into another operators network with the same identity, be authenticated based on his home identity and the circle-of-trust, represented by the roaming agreements, established between his home and his host networks.

Despite the fact that Operators already play a role of Identity Provider for its users, the extensions of this role as an IdP towards other Service Providers can greatly add value to the Telecom Operator business, as evidenced in (GSM Association, 2008). This would allow:

- More services to be launched for the user through the operator as a trust channel.
- User-friendliness towards services access, through SSO and federation agreements.
- Content personalization by tailoring the data towards the user based on the attributes of his identity.

At the same time, there are several advantages in hosting ID's for other service providers in the UICC:

- The UICC already carries an identity towards the Operator.
- The UICC is a secure environment for the storage of the keys, biometric data, application of cryptographic operations.
- The UICC is widely deployed, has a great customer penetration; it is pervasive and a necessary component in the mobile communications.
- It is defined by international standards, and there is work in progress towards internetworking the 3GPP specifications for the UICC with the Liberty Alliance Identity Management Standards (3GPP Technical Specification Group Services and System Aspects, 2008).
- The UICC can be multi-application, and it is device independent and interoperable. Moreover, it correspond to an user credential in respect to "what he has" and can be used in conjunction with "what he knows", by having a PIN or password, or with "what he is", by making use of biometrics.
- The support of multiple security domains defined on the Global Platform Standards, enables the operator to act as a "Real State" provider by offering a secure and isolated environment to service providers.
- As discussed in the thesis, the size and communication interfaces limitation of the SIM are being broken by the new technology advances.

The previously mentioned national E-Id smart cards initiatives introduces neat approaches towards Identity Management such as allowing both public and private applications residing in the same card, or the seamless implementation of “alias” in the Austrian E-Id example. However those are somehow local initiatives, and in order to deploy a more global framework around the SIM, the GSMA has publish an IdM Framework, (GSM Association, 2008).

The GSMA IdM Framework proposes breaking the role as Identity Provider in both Authentication Provider and Identity Attribute Provider, although both roles could be played by the same entity. The first one validates the user's credentials and based on that provide authentication assertions about the user. Whereas the second one facilitates the exchange of the user's attributes between trusted parties.

The operator can have one or both of the roles, or have a third party provider as an IdP using the "Real Estate" security domain in the SIM. It could even have both cases implemented on the card as it can have multiple identities.

If another entity acts as the IdP, the Operator must create the security domain and make the handover to the IdP. So the IdP can manage it, by: providing the credentials to the users, being able to revoke them and being responsible for the agent that will use the credentials towards the Service Providers. Where, the Service Providers are the ones that require the user authentication and/or may use user information through the Identity Provider. An example of that can be seen in the Figure 15.

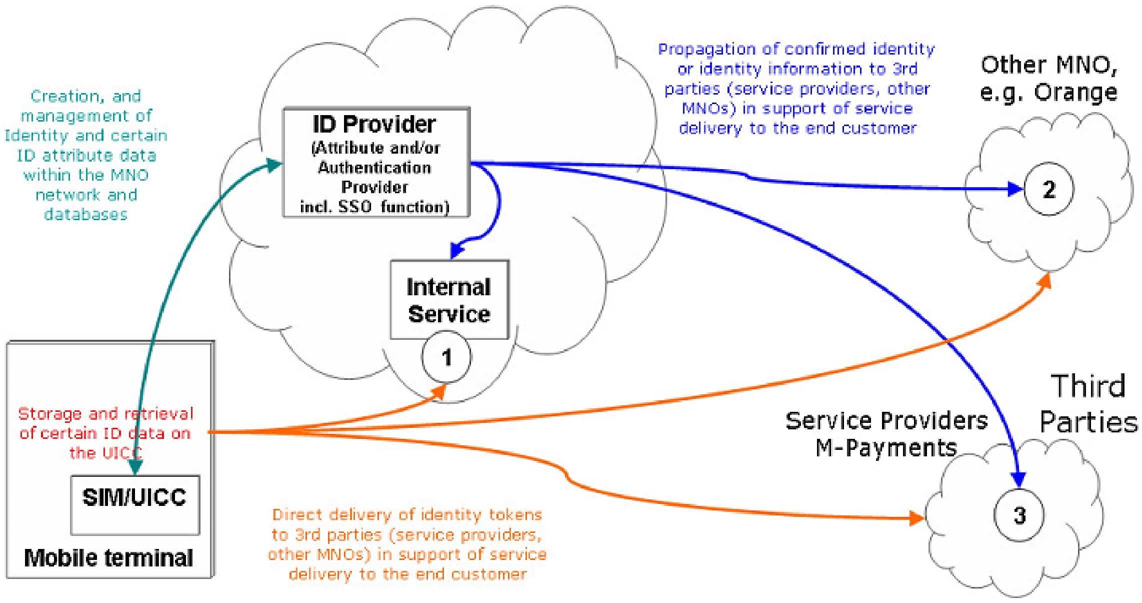

Creation, and management of Identity and certain ID attribute data within the MNO network and databases
ID Provider (Attribute and/or Authentication Provider incl. SSO function)
Internal Service
1
Propagation of confirmed identity or identity information to 3rd parties (service providers, other MNOs) in support of service delivery to the end customer
Other MNO, e.g. Orange
2
Storage and retrieval of certain ID data on the UICC
SIM/UICC
Mobile terminal
Third Parties
Service Providers M-Payments
3
Direct delivery of identity tokens to 3rd parties (service providers, other MNOs) in support of service delivery to the end customer

Figure 15: UICC IDM Architecture example from (GSM Association, 2008)

In this IdM architecture, the users register themselves towards the Identity providers, feeding them with data, either for the subscription or also dynamically providing attributes, that could acquired in real time by sensors. They also share a proof of identity with the provider, in order to obtain an authentication. The method to do it can be through username/password, certificate, shared key, PKI, etc. At last, the user is able, based on an agreement towards the ID provider, to manage his privacy and policies for information sharing about him.

The SP needs to be able to discover the user's IdP, so it can validate the user identity, and as well, establish a trust relationship with him. The same trust relationship allows the SP to receive attribute information linked with an identity, in order to customize the service. At last, it should be possible for the ID provider to supply information about the accuracy and trustworthiness of the assertion and to communicate the data to the SP by the usage of pseudonyms in order to maintain the user privacy.

(GSM Association, 2008) sets some security requirements for its framework such as ensuring a limited lifetime to the tokens, conveying risk assessment data to the assertions, combination of the ID information with attributes, Single Sign On (and Off) capabilities, support of federation and multi-domain interoperability and alignment with open standards as it endorses the convergence with the Liberty Alliance standards. Besides that the, framework points the UICC as the identity host.

Even though the UICC already offers a proof-of-identity towards the operator, it should be able to provide a different proof-of-identity to the Service Provider, once the SIM/USIM/ISIM authentication is valid only in the Operators domain. As mentioned, one alternative is to generate and store, in the UICC, an authentication token such as a SAML token or a X.509 certificate complying with both ID provider and SP, making use of the SIM's Real Estate Capability. The token can be provisioned either before issuing the card or afterwards through OTA, based on the global platform standards.

Another aspect detailed in the GSMA IdM Framework is the attribute supplying capability that could be offered through the identities. An Attribute provider in the mobile context could offer information such as: user age, presence, location, platform, preferences, historic and any other information that could be retrieved either from his subscription, or from sensors in his mobile, or even through relationships established inside a service but shared with his identity. The mobile as a pervasive equipment that is often carried with the user and which is aggregating more and more uses and sensors is a perfect tool for acquiring user attributes, and the SIM serves as the security factor to protect this information. An example of an attribute providing case is shown in Figure 16.

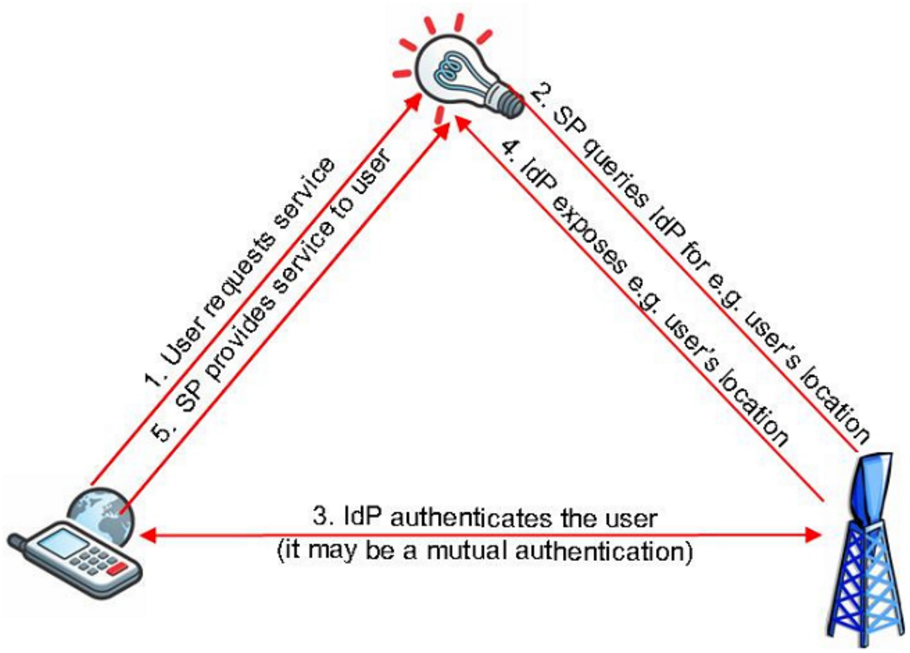

1. User requests service
5. SP provides service to user
2. SP queries IdP for e.g. user's location
4. IdP exposes e.g. user's location
3. IdP authenticates the user
(it may be a mutual authentication)

Figure 16: Attribute providing example from (GSM Association, 2008)

The attribute retrieval could be done in 3 different models:

- Pull-model: The SP requests a specific attribute from the Attribute Provider.
- Push-model: The Attribute Provider autonomously feeds the SP.
- Subscription model: The SP subscribes to updates of specific attributes and is fed when those attributes change.

This involves a somehow complex setting of privacy policies in between the identity provider and the subscriber and a circle-of-trust establishment between the IdP and the SP. However, the Telecom Operator seems a good candidate to do so, as it already works as an IdP, it has SLAs signed with Service Providers and the user most likely need to sign a contract with him anyway. It is also worthy to mention that the inclusion of the use of the open standards SAML and WS-* in the GSMA IdM framework represents interoperability with other industry IdM standards.

Moreover, the security domains on the SIM and the multi-application cards confirm the trust of several identity providers and service providers in the SIM as a secure identity platform. A highly connected identity platform which, for example, could use its context-awareness capabilities to select identities transparently. Or, it could feed Attribute Providers with real time attribute data.

7. Trust

In this chapter we will review the concept of trust describing its characteristics and we will explain the two approaches for trust modeling: the policy-based and reputation-based models. Later on, we go deeper into the modeling strategy and trust measuring for the reputation-based model, once the policy-based is already handled, up to some extent, by the SIM. At last, we present a small discussion on social networks showing the trust and identity aspects of it.

7.1. Definition

There are several different definitions of trust. Here it is some examples we found during our studies:

- "Trust is the belief that the *trusting agent* has in the *trusted agent*'s willingness and capability to deliver a mutually agreed service in a given context and a given time slot" (Chang, Hussain, & Dillon, 2006).
- "Trust (or, symmetrically, distrust) is a particular level of the subjective probability with which an agent will perform a particular action, both before [we] can monitor such action (or independently of his capacity of ever to be able to monitor it) and in a context in which it affects [our] own action" (Abdul-Rahman & Hailes, 2000).
- "Trust of a party A to a party B for a service X is the measurable belief of A in that B behaves dependably for a specified period within a specified context (in relation to service X)" (Artz & Gil, 2007).

Although the definitions may change a little bit, all of them include the relationship between at least 2 agents and some belief/reliance. In this thesis, we will use the definition given at (OASIS Web Service Secure Exchange TC, 2007), where "Trust is the characteristic that one entity is willing to rely upon a second entity to execute a set of actions and/or to make a set of assertions about a set of subjects and/or scopes." Thus, trust can be a relationship between two individuals, one individual and a company, one individual and a product, one individual and a statement, etc.

The motivation behind the study of trust is that it corresponds to the input to judgments of problems that can not be easily solved by direct logical analysis of known facts. This kind of situation is often more common due to information overload and increase of dynamics for taking decisions.

Trust is different from security, although it is pretty related to it. In order to establish a trust relationship, sometimes it is necessary to use security mechanisms such as digital identity, signatures, encryption, etc. In the other hand, the usage of those mechanisms depends on the parties to trust that they will in fact secure them. Trust is based on a belief; it indicates an expectation; it does not provide any guarantee of a successful outcome.

7.1.1. Factors Affecting Trust

Trust is context dependent. It is easier to see that in the case of trust between two individuals. Bob, for example, can trust Alice about health topics but not about computer-related topics. Besides the fact that there may be an enormous amount of different contexts, the users interacting may have different views of the context itself. If someone is looking for a security expert, would an Internet security expert be enough? In this case it may seem like a problem on detailing the specification, but it is not intuitive how much detailing is necessary to assert trust, especially if these trust it to be forwarded to other users. Another problem may happen due to misconception between the context specifications, as a user may just have a wrong definition of a concept. Some literature about trust does not take context into account due to the complexity inherited by that dependency. However, if the context is abstracted in the trust modeling, the trust scope can be seriously limited.

Those problems happen because trust is fuzzy and dynamic. As mentioned before, it is a belief and people have different believes and they base those in different factors, besides giving different weights to each factor. The weight of the trust relationship may be different for each party, and an event will affect each agent's trust differently. For example, for Bob to offer a wedding ring to Alice may not correspond to an as big step in the mutual trustworthiness as it may for Alice. This belief difference also determines the asymmetry of trust. One user may trust another, but this is not necessarily reciprocal. It is not because Bob trusts Alice to borrow his car, that Alice will trust him for borrowing hers.

When it comes to trust between people it should be considered both the willingness and the capabilities of the agent to be trusted, as mentioned in (Chang, Hussain, & Dillon, 2006). One will trust the other in a certain context due to his expertise on that context, but also on his disposition in providing true information. The same way Bob would not ask Alice about which computer to buy due to her lack of knowledge on the area, he may not ask a representative of a computer manufacturer once he may provide biased information.

Trust is also time dependent; it is dynamic. The reason why time is so important in trust is because the willingness and capabilities of a *trusted agent*, as well as the familiarity between agents may vary over time.

Trust is built based on the knowledge from the *trusting agent* about the *trusted agent*. This knowledge may be based on previous direct interactions between both of them, by recommendations from other agents or by reviewing historical data about the *trusted agent*. It may even be supported by legal agreements such as SLAs.

The fact that we build our trust on these external input (other agents' recommendation, past data provided by a trusted source) is based on trust transitiveness. The transitiveness is one of the most important aspects of the trust. Before a first trust interaction, all the *trusting agent* can rely on is what he learns from the experiences of other people.

7.2. Policy-Based and Reputation-Based Models

In the case where there is no previous direct interaction between the agents, the trust may be estimated by asking the opinion of 3rd party agents or assessing historical information about the *trusted agent*. In both cases, it relies on an already established trust relation between the *trusting agent* and the 3rd party agent or the information source.

We can clearly observe the trust concept in the identity and attribute claim support at the Identity Management models described in the previous chapter. A user registers him, developing a relationship, with an IdP. Due to this relationship, the IdP is able to prove some of the user's claims, such as identities and attributes. The IdP acts as the 3rd party providing trust to the user's claim. The strength of the model is based on the common trust on the IdP and its mechanisms to establish that relationship and to be able to prove the claims. You can take this example to the physical world where for example when trying to prove your identity, your sport club card with your picture is enough so that the club staff trust you to be a member and that you are yourself but it may not be enough for a policeman to trust that you are above 18.

This relationship between the user and the Trusted Authority (in the previous case, the IdP) is often based on SLA or contract. The IdP may have to acquire data from the user (through the registration for example) at a reliable process (by checking the user national ID to be sure that he is an adult and etc) so he can be able to make assertions about the user and so other people can trust his assertions.

The mentioned example above is what is described in the literature as policy-based trust (Artz & Gil, 2007). It relies on the security behind the agreements of the Identity authorities which are enforced by certifications, auditions and SLAs. The trust result is a binary trust or not-trust to the claim.

However, trust can also be extended to reputation systems, where entities have their reputation rated by direct or indirect past interactions; interactions that do not have a relationship built through SLA or policies. This is of great usage in scenarios where it is important to generate trust over claims that are somehow subjective or more context-dependent. Some examples of those claims would be: “is this an interesting article?” - it depends for whom; “is the staff of that restaurant friendly?” - it relies on a personal opinion. A policy-based trusted authority would not have enough mechanisms to prove the answer to those questions due to the fact that it is not possible or feasible to accumulate hard evidences towards it.

We will discuss both approaches in more details both approach in the next subsections.

7.2.1. Policy Based Model

Both policy-based and reputation-based trusts rely on the recommendation from a 3rd party. The main difference is based on the relations between this third party and the agents that it is providing trust assertions. In the policy-based, the relationship is based on policies and SLAs; it is usually endorsed by hard evidences that can be proved. The nature of these relationships is usually protected through cryptographic algorithms and protocols. The 3rd party in this case is known as a trust authority.

One good example of policy-based trust is on the PKI (Linn, 2000). In the PKI, the users believe on the certificates from each other based on the fact that they are signed by a trusted

CA, or a Certification Authority that has a trust relation with a trusted CA (cross-certification).

The Trust Authority must care about his reputation in order to have the user's trusting him and to be able to establish mutual trust with other Trust Authorities. This responsibility around the trusted authority pushes it to develop evidenced-based methods (such as physical registration process) to prove claims that it will be responsible for. For example, in the case of an entity that acts as an Identity Attribute Service Provider, it should enroll the user into an unbiased process in order to assure that those attribute claims are in accord to the policies understood by the users or systems that will retrieve them from this provider.

Besides securing the process of acquiring the data to be used in the claims (either attribute or identity claims), the Trusted Authority should protect the data dissemination in order to maintain its reputation. If someone impersonate one of the trusted authorities' claims or alter the content of the claim, the authority will lose its credibility. This is the main driving force for using cryptographic primitives such as digital signatures when involving trusted authorities.

The policy-based trust may involve a cost, once the trust authority has to invest capital in order to correctly acquire and maintain the data to be proved, to ensure its validity (some data are dynamic) and to be able to offer an unbiased point-of-view. This model seems more feasible (and has been more widely deployed) in contexts where the veracity of information has a great impact on the business such as in e-government, or in e-commerce between companies and customers.

The policy-based approach is already implemented in smart cards and SIM cards that carry PKI certificates. Moreover, the case of the MSISDN and IMSI identity towards the operator, and the roaming agreements towards other operators as another complex case of a SLA ruled policy-based trust. More on the policy-based approach can be understood from the discussions around Identity Management models presented on the chapter 6 due to its shared aspects between the policy-based trust models.

7.2.2. Reputation Based Model

The reputation based systems are built through shaping the trust based on the past interactions and combining those interactions in order to formulate the trust indication (Artz & Gil, 2007). Due to the limitation of interactions that one agent can have, those systems normally employ trust reputation transitivity.

By having every individual taking part on the system and being able to create or support a claim, all of those *recommendation agents* act as a kind of trust authority. And, the problem of maintaining its reputation is distributed to all the users on the system. Nevertheless the weight of the judgments is also distributed to all agents acting as *recommendation agents*.

This mechanism can rely on a system or entity (possibly a Trust Authority) that offers an identity to the user so he can act as a *recommendation agent* and which offers the IT infrastructure for those users to play that role. The users can establish digital relationships between themselves based on long-term relationship such as friendship or less established relations such as being engaged in a operation that evolved trust (such as an online purchase) and everything was ok. Those long-term or short-term relations ends up to build a trust network, a web-of-trust, somehow similar to a federation in the case of the policy-based Trust Authorities (but loosely as it is not based in contracts).

Nevertheless, the whole mechanism could be decentralized, with every agent storing his trust information, his personal weights and defining his transitiveness interfaces. It would be only necessary to follow the same protocols and trust definitions in order to be able to forward the trust in between themselves.

The reputation systems can be built so just the user reputation or claim rating is widely available without any information of who rated who, representing a global average evaluation. Or it can provide a more complex interface by keeping the correlation between the claims and the supporters or even giving a different weight if the trust target has a direct or indirect relation with other party. The reputation rates can be stored and published in a central system or remain at each agent's system.

The fact of not having the strong protection of SLAs or contract when sharing a trusted information makes this model more suitable for categories of services where the veracity is not that crucial or is relative/subjective, or where the relationship requirement are more loose. Some examples are: opinion and activity sharing web sites, social networks, job networking web sites, etc.

However, as shown in (Dellarocas, 2003), reputation systems built over individual on-line feedbacks manage to provide reasonable trust enabling systems that seems really risky such as on-line auctions to achieve great adoption and usage. Although, they often implement some escrow over the ratings and recommendations. In general, this kind of reputation based information is being more and more accredit and now represent a big weight in user's decisions varying from choosing a music album to buy or which company to invest. Due to the fast dynamic of today's world and the widespread of information, the lack of knowledge in some decisions is being compensated by information gathered through trust relationships. This can also be extended to be applied in the Semantic Web to automate judgments done by agents and systems.

A good analysis on the trust systems mentioned above and reputation systems can be found in both (Massa, 2007) and (Chang, Hussain, & Dillon, 2006).

7.3. Modeling the trust

We will focus the trust model discussion over the reputation based approach, since it is the more complex one and it is not ruled by mature standards or frameworks. The trust modeling for the policy-based approaches are ruled by the SLAs or contracts towards the Trusted Authorities and between those authorities (varying mainly in questions regarding mutual trust and federation chaining).

Since trust corresponds to an inference, a belief, it is natural that it will be modeled based on the composition of the input of several sources that can also change through time. Those sources can be gathered both through direct contact (including the historical data of it) or by collecting information from the experience of other peers. In addition, there is the personal user own inclination towards trusting other agents. We graphically represent trust as the aggregation of those components as illustrated in the Figure 17.

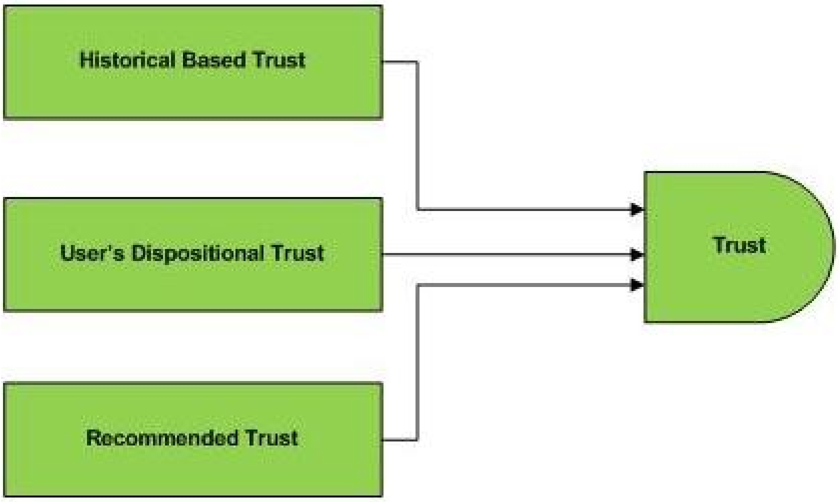


Figure 17: Trust Components

7.3.1. Dispositional Trust

Some models also take into account what is called the dispositional trust, or "basic" trust. This concept corresponds to the default trust behavior of the *trusting agent*, and it is often modeled as an optimistic or pessimistic behavior.

The optimistic behavior would be to accept to trust others unless proven the opposite. This is more suitable for situations with low risk and with big global benefits on cooperation. The pessimist one is the opposite. The agents do not trust each other unless there is a reason to do it, consequently, it is an approach proper for decisions of higher risk and where the agents do not have much incentive to cooperate (Alani, Kalfoglou, & Shadbolt, 2004).

This behavior is materialized in the trust model through the assignment of the trust value based on an interaction between the agents, or on the modeling and weighting of the rules or equations that will result on the trust. The decision between each one of those behaviors, particularly when the trust is public accessed, is a topic discussed at the Game Theory (Sanfey, 2007).

On what concerns dispositional trust modeling, another approach may be to let the users attribute those weights or to try track the user behavior and adapt the weights in accord. The first strategy requires a significant effort in human interface and psychology in order to capture the user disposition and embed it on the trust algorithm. The second one requires artificial intelligence work towards machine learning.

7.3.2. Historical Based Trust

The second component of trust described here is the historical based one, which is sometimes referred as direct trust. This component is derived from the summary of the interactions between the agents. It is a fundamental part of the trust, once if there are no records of previous interactions; no recommendations can be exchanged between agents.

Besides the challenges to model it due to the context sensitiveness and dispositional influences, an important aspect in its modeling is the time degradation of trust. In order to be able to employ it, a database or similar must be built storing each one of the interactions and when they occurred.

Trust deterioration due to time is important to be considered once agents and evaluation criteria for trust evolve. A possible approach would be to redo the past trust judgments when they need to be accessed and use the current evaluation criteria of the *trusting agent*. Nevertheless, it could be too cumbersome depending on the number of judgments and how old they were. The most common approach (Caverlee, Liu, & Webb, 2008), (Chang, Hussain, & Dillon, 2006) is to assign a time weight to the trust value of the past experiences, where this weight is decreased with the passage of time.

Before adding the transaction to the historical database, it is necessary to evaluate the trust result of it. This trust value represents a confrontation between the expected result and the final result of the interaction between *trusting* and *trusted agents*. It often corresponds to something not trivial to describe. Nevertheless, (Chang, Hussain, & Dillon, 2006) suggests the prior definition of quality aspects and quality assessments criteria to be agreed before the interaction in order to get values with less subjectivity. An ontology representation of this direct trust is presented in Figure 18. This tactic is used by several services for rating products on the web, such as: CNET[54], Bizrate[55], Yahoo![56]. The problem is that the definition,

[54] http://www.cnet.com/
[55] http://www.bizrate.com/
[56] http://shopping.yahoo.com/merchrating/general_info.html

agreement and assessment of those criteria make the assessment more complex and hard to be ported to more diverse and seamless types of interactions.

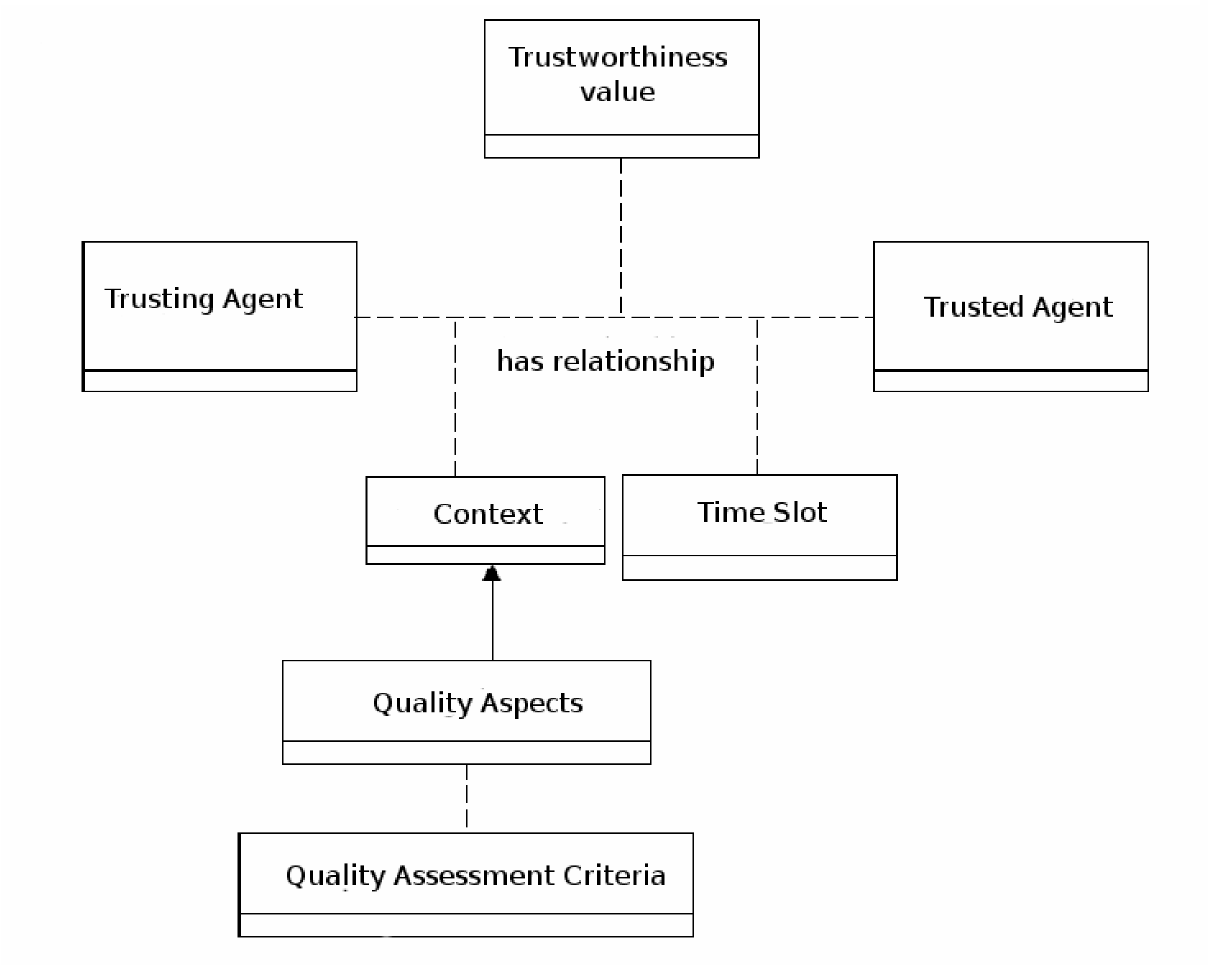


Figure 18: Generic Trust Ontology from (Chang, Hussain, & Dillon, 2006)

7.3.3. Transitiveness and Recommended Trust

Before discussing more about recommendations, it is important to clarify the different roles in the reputation model. The agent who is inferring the trust is the trust agent. The one whose trust is being assessed is the *trusted agent*. And finally, the one who is being consulted (whose recommendation is being asked) is the *recommendation agent*. The ontology represented in Figure 19 and presented in (Chang, Hussain, & Dillon, 2006) illustrates the relationship between the three different agents.

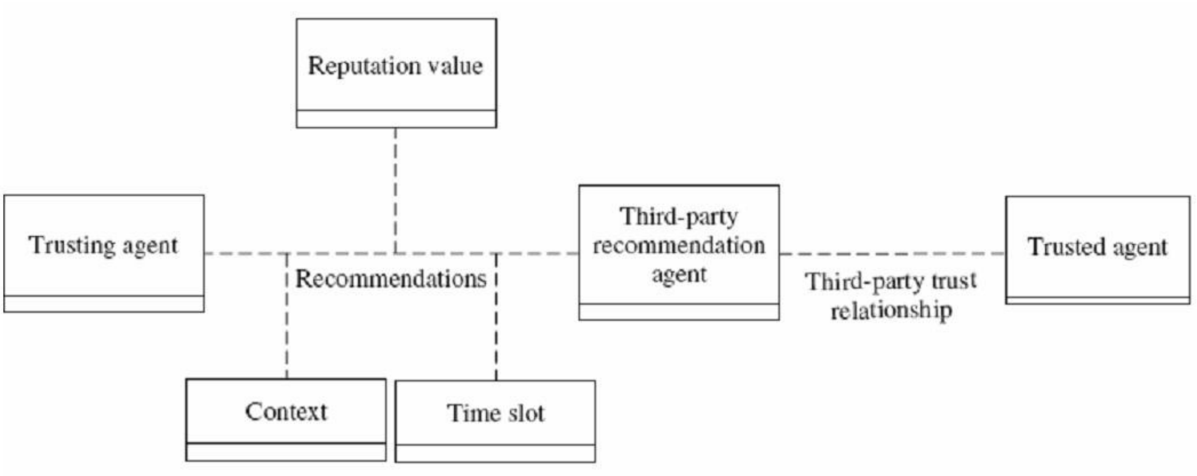


Figure 19: Reputation Ontology from (Chang, Hussain, & Dillon, 2006)

Transitiveness is one of the aspects that the researches pay more attention when modeling trust. It is a logical approach, as before doing a judgment about a certain context for the first time, no historical data can be of use. Transitiveness is something to be explored carefully. In one hand, following the theory of the six-degrees of separation (based on the studies of Stanley Milgram in (Milgram, 1967)), we can easily find a source of information to solve a trust question. On the other hand, the further it is that source of information, the less we can rely on it. Modeling transitiveness is very challenging, as trust is not perfectly transitive and it degrades both with time and as it passes through the transitiveness chain.

When the recommendation or assertion goes through the trust chains, the final recommended trust decreases at each pair due to the trust uncertainty between the nodes. This effect of lost of trust reliability is due to the individual interpretation of each node, besides the uncertainty on their mutual trust. A simple example of this effect can be perceived in the game known as "Telephone". In this game, the first player whispers a phrase to the next one, which propagates successively to the next one, until the last player is reached. Then, the last player announces the statement and they observe the amount of errors in the final message.

Variants in the transitiveness models occur based on how many distance degrees are taken into account for the trust assembling. The model proposed by (Taherian, Amini, & Jalili, 2008) for example does not limit the size of the chains to be considered, although it does add a deterioration factor to represent the trust chain degradation. In the other hand, in (Ann

Golbeck, 2005), the authors limit the trust transitiveness to the shortest path, and other models confine the chaining in just one or two nodes away.

It is important to notice that the trust between the *trusting agent* and the *trusted agent* is a different one from the one between the *trusting agent* and the *recommendation agent*. Both trusts refer to different contexts. One is the trust on how good the agent is as a *recommendation agent*, while the other is the trust on the agent itself towards a specific context. Thus, the trust models that take both trusts into accounts represent them as distinct relationships in the trust database or trust ontology, and with different value ranges.

7.3.4. Modeling Transitiveness

This trust on the *recommendation agent*'s ability to give a beneficial advice is as dynamic as the trust between *trusting agent* and *trusted agent*. Thus, models such as (Abdul-Rahman & Hailes, 2000) make use of a database for recommendation experiences in order to later be able to tune this trust value between recommender and *trusting agent*.

An interesting approach taken by (Taherian, Amini, & Jalili, 2008) was to model this transitiveness loss through a resistive network as an analogy to the representation of an electric circuit diagram with resistances and diodes. The parameterization of the trust loss during the network path, as a resistance in the case of (Taherian, Amini, & Jalili, 2008), can be based on both the distance between the edge nodes and the mutual recommendation trust values between each pair of nodes in the chain.

As mentioned before, challenges arise around the transitiveness due to subjectivity of the recommenders trust and the difficulty in contextualization expression. Those challenges make it extremely complicated to model a trust value as a probability measure (Abdul-Rahman & Hailes, 2000).

For quantifying the trust into the recommending agent, some models choose a binary approach such as good/bad (Caverlee, Liu, & Webb, 2008), while other user a graduate scale. Some consider just positive levels of trustworthy (Ann Golbeck, 2005); or both positive and negative levels; and they may or may not consider the case where the level is unknown. For example, the recommender's trust has been modeled as {very bad, bad, unknown, good and very good} in (Abdul-Rahman & Hailes, 2000) and as a 5 start scale in (Chang, Hussain, & Dillon, 2006)

The realization of the transitiveness imposes the communication of the recommendations between the agents. This can be facilitated by a central system that stores those values and then, the users just query this system to obtain the recommendation values. A model based on centralized trust systems is explored in (AlNemr & Meinel, 2008). Otherwise, it needs the user to trigger a recommendation broadcasting (to be propagated trusted network chain) every time it wants to infer some trust.

7.3.5. Modeling the inter-agent trust

Due to the different trust components and to their relative complexity, the inputs and weights for the final inference of the trust in between *trusted agent* and *trusting agent* vary significantly in between all approaches observed during this study. This complexity must also be considered before the development of the system once it offers a trade-off between usability, development time and cost against accuracy of the system.

The inputs are usually stored in a database and fed into trust measure calculation such as a set of deterministic equations, a fuzzy system or a Bayesian model. Thus, the result of the calculation could be either a decision to trust or not the *trusted agent* or to attribute a value (that could be binary or scalar) to the relation. Then, this value can be used by the user to decide upon the new interaction.

As an example, (Chang, Hussain, & Dillon, 2006) presents a set of deterministic equations for calculating the trust value in relation to service providers and to aggregate recommendations from *recommendation agents*. In order to calculate the mutual trust, it uses a set of quality of service criteria that are aggregated and weighted based on how clear are the criteria and how much does it influence the final trust. For the transitiveness, they use the mutual reputation of the agents, the distance between the agents in the social network and a time degradation factor in order to weight the recommendation from a *recommendation agent*.

In the other hand, (Schmidt, Steele, Dillon, & Chang, 2007) defines a set of fuzzy rules in order to determine the trustworthiness level between agents. They use as input a weighted trustworthiness value that was received from a *recommendation agent*, besides the weight of that *recommendation agent* into the *trusting agent*'s opinion and the credibility of the *recommendation agent*.

In any approach, the final trust value can be contextualized, as having a value for each context (in different context databases) or generalized. As (AlNemr & Meinel, 2008) mentions, the majority of the current reputation systems do not differentiate between general reputation and contextual reputation. For a more seamless approach, this is an issue because the trust relation between users is very dependent on the context.

The approach mentioned so far models this trust value as a representation of the belief but not the certainty on this belief. (Jøsang, Hayward, & Pope, 2006) proposes a model based on subjective logic, where the belief and the confidence are inputs to the model. Instead of having the sum of the belief and disbelief as 100% of probability, the subjective logic defines a base where the sum of belief, disbelief and uncertainty correspond to 100%.

7.3.6. Global trust

So far we have been mainly discussing the cases where the trust is a private value that belongs to each user as many-to-many relation between *trusting* and *trusted agents*. This is called in the literature as local trust. A different approach is to have this trust value as a global value.

In the global trust, the distance and type of link between the *trusting agent* and *recommendation agent* is not so important. The trust value of the *trusted agent* is calculated on the recommendations of the whole network and made it public. This global calculation is performed by a central system that often have some moderation capabilities, such as banning fraudulent users, spammers, etc. Practical examples of usage of those systems are online sales such as Ebay[57] or Amazon[58].

[57] http://www.ebay.com/

[58] http://www.amazon.com/

Some of those systems add mechanisms to praise active users and stimulate that they keep evaluating the agents' trust. One of this mechanism used in Epinions.com[59] is to promote the user's trust level to a different category, such as rating him as an expert (besides his global rating voted by the system users) if he is very active in a context and receive a certain amount of positive feedback. Other systems such as Yahoo! Merchant Ratings[60] provide feedbacks from experts outside the system besides the recommendation of the recommending agents. As mentioned in (Chang, Hussain, & Dillon, 2006), users often pay more attention to those more credible recommendations then the global average. The choice between relying on local trust or global trust can be based on the user's dispositional trust towards the reputation context.

The systems of global ratings are subject to cases where a user acts honestly for a while in order to achieve a good reputation and then intentionally add a biased or false recommendation. The solution to that case is to keep track of the user's behavior and to add this behavior's change as an input variable in the global rating, as done in (Caverlee, Liu, & Webb, 2008).

Another issue noticed towards global trust is that the publication of trust values can make the users feel compelled to give false ratings or accepting unwanted connections due to fear of offending the other party. This is an important issue which can represent a great threat on the efficiency of the system and it should be considered when opting for using local trust values or global trust values.

7.4. Online Social Networks and Trust

Social Networks are closely related to trust. The real social networks correspond to groups of people with something in common. We can characterize our families, our co-workers, social groups or others as social networks. This common interest between those people inside the social network enables them to share some extent of trust in it.

Due to the widespread usage of the internet, social networks started to be mirrored, developed and materialized in the virtual world. The trigger for this was the usage of the internet for

[59] http://www.epinions.com/

[60] http://shopping.yahoo.com/merchrating/general_info.html

connecting not only the computers but people. As mentioned in (Breslin & Decker, 2007), e-mail, mailing lists and bulletin boards enabled people to create the first social networks. Although not labeled as social networks, people, through those means, shared a common interest in a topic, engaged into discussions about it and were able to share their.

Recently, websites such as Facebook[61], Myspace[62], Orkut[63] and Linkedin[64] made explicit those relations between members by having the user to link him with the other users. Another new concept was the user profile, which represents a digital identity of the user on that community (by having the website system as their Identity Provider). Those profiles are aggregating more and more information about the user, although they are hardly supported by any "hard evidence".

The Online Social Networks (OSN) profiles can contain information such as the user real name, pseudonym, birthday, location (both residence and "real time" location), religion, personal interests, personal background (such as work and academic experience), contact information and content that the user publish in the web (such as pictures and videos). As reviewed in (Dwyer, Hiltz, & Passerini, 2007), the OSN members reveal a lot of information about themselves.

As pointed by (Fogel & Nehmad, 2009), the availability of information on those social networks leads to some privacy concerns due to users seeking both control over intrusion (for reasons such as avoiding behavioral response and evaluation by others) and control over disclosure (for protection of identity and self-image). However, studies such as (Stutzman, 2006) and (Fogel & Nehmad, 2009) show that, in general, users are not that much concerned about access to their data by strangers and they were very positive about sharing this data with their friends.

Besides the profile creation and relationship building, the Online Social Networks offers creation and discussions of domain related subjects, profile surfing, message exchange and publishing of content (such as videos, pictures, blogging, etc). Some social networks are more general while others are more context specific based on its privacy settings, user interface,

[61] http://www.facebook.com
[62] http://www.myspace.com/
[63] http://www.orkut.com
[64] http://www.linkedin.com/

profiling and publishing options. For example: Linkedin focus on professional networking as most of the content published is business/work networking related and your profile corresponds to a curriculum vitae. In the other hand, Last.fm[65] is music oriented and it senses the user musical taste based on what songs he listen to. Through that, it builds his profile and recommends groups and users with similar musical taste. MySpace and Facebook are more contexts free or generalist networks although they allow the creation of groups of interest.

The popularity of OSN is so big that about 20% of the WWW page views are from the social networks MySpace and Facebook, where this corresponds to half of the view between the top 10 most popular domains (Breslin & Decker, 2007). Facebook has more than 200 million active users where half of them access it at least once a day (Facebook, 2009); some older data source concerning MySpace reported more than 110 million active users in it (Techradar, 2008). Those two Social Networks together have more active users than the whole population of the United States, the third most populated country in the world.

Still, despite this complexity around social networks and their popularity, their whole potential is somehow underused. They have a lot of information about the user profile and about his relation, but there is a general lack of contextualization and characterization of this data. There are just few mechanisms to rate or describe the relations between users and groups in OSN. Besides, the majority of users do not use those mechanisms.

One of the reasons for the lack of trust in OSN, as mentioned in (Breslin & Decker, 2007), is that many social networks lack somehow of a common objective, and, as a consequence people connect to each other for only boosting their number of connections. Moreover, several users feel compelled to accept friendship invitations despite they would not in the real life (Beattie, 2005). A survey done with some users of the Orkut social network showed that about one fourth of the connections that those users have done was due to a feeling of an obligation, as users preferred to add an unwanted friend instead of possibly offending the person (Ann Golbeck, 2005).

This clearly points to the fact that the current Online Social Networking model does not really represent a trust relation between users, although it could. A study (Dwyer, Hiltz, & Passerini,

[65] http://www.last.fm

2007) comparing MySpace and Facebook (while it was still coupled to physical entities such as universities) shows the weaker the trust represented in the OSN the more the people were building online relations despite not knowing the other user. Thus, the issue is more oriented on the lack of categorization of those relations that ends up representing the same value in a close relationship and a loose relation over the same network. This is especially dangerous when those relationships are subjected to the transitiveness of the OSN. This lack of trust is even enabling the development of scams (Elgan, 2008), SPAMS and crimes (as kidnappings) based on weak trust relations developed on online social networks.

Some of the online social networks do have some mechanisms to weight and define the relationship between the users as well as setting the users privacy based on those. Nevertheless, this process is not carried by most of the users. Judging by the resources offered to categorize and attribute trust indication ratings to friends, we would say that the problem is on the fact that this is not done seamless. Defining the relations in the current OSN demands the users to active interact within the OSN and engage a lot themselves.

Another underuse noticed on OSN is due to the lack of integration between different social networks. Their context particularities leads the users to create profiles to different social networks and aggregate a lot of information on those, that theoretically could be used to both determine and attribute a ranking on its trust capabilities or willingness for several different contexts, but this information is not shared. The users develop several relations between each others, and share several interests, but those relationships are fragmented in the different OSN. The Figure 20 greatly illustrates these relations between the user’s different identities and content.

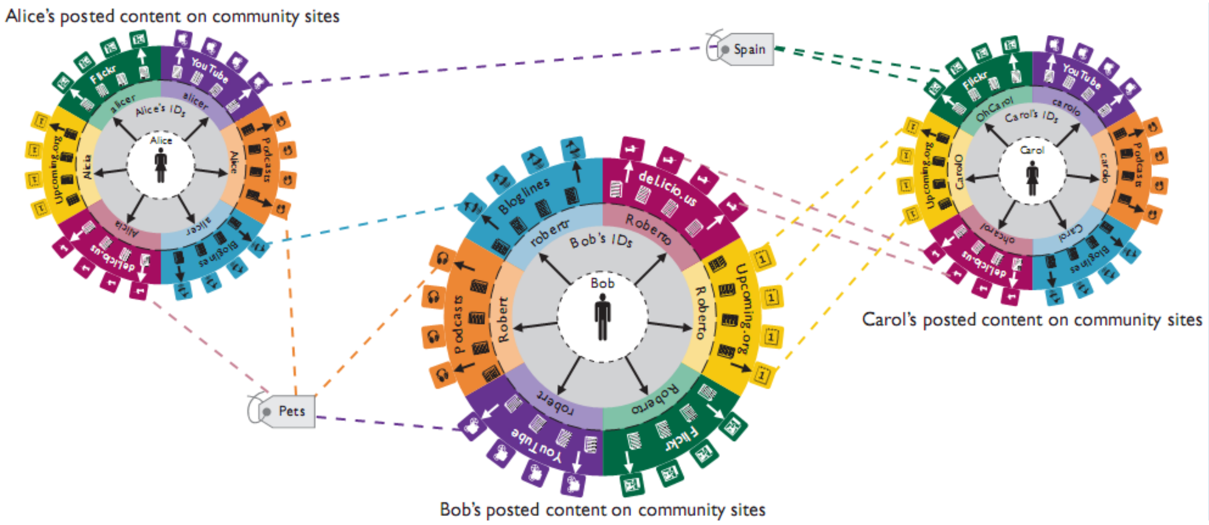

Alice's posted content on community sites
Spain
Flickr
YouTube
Podcasts
Bloglines
del.icio.us
Upcoming.org
Alice's IDs
Alice
Bob's IDs
Bob
Roberto
Robert
Carol's IDs
Carol
Pets
Carol's posted content on community sites
Bob's posted content on community sites

Figure 20: Users relations based on their different accounts and common interests from (Breslin & Decker, 2007)

Some work has been started on providing service oriented access to their online identities, to export the information so it can be inserted in another social network without having the user to type all his data (Six Apart, 2007) or to create a common RDF/OWL vocabulary for describing users and their interconnections (Brickley & Miller, 2007). Nevertheless, those actions are still somehow at an initial stage.

A very interesting approach suggested by (Breslin & Decker, 2007) is to move the social networking into a stack below the internet applications, rather than having it as an application itself. Then, the information aggregation due to the identity profile can be reused and the relationships can be easily managed. This strategy of having the social network as a low level stack for application is somehow already in use in Facebook and Orkut where it is possible to develop applications for those social network platforms. Still, a common and platform independent social networking stack would be much more useful. In that perspective, a trusted, personal and seamless device such as the SIM card could fit very well.

Having said that, there is a great potential for using social networks to infer and represent trust. For the problem of information fragmentation, open standards and an attribute provider framework as the one described by the GSMA IdM framework or as a social network underlying application layer, (Breslin & Decker, 2007) could be the solution. However, those networks still seem to miss a mechanism to assert the real trust value that represents the relationship based on both relationship context and the real (not virtual) relationship between the users. They lack some hard evidence based mechanism to endorse the trustworthiness of the data that they host.

In the next chapter we present how the future SIM can tackle this need of providing a real trustworthiness value to the relation.

8. Trusted Service Design

In this chapter we will summarize the trust enhancements which the Future SIM could provide to the services around it, discuss the possible application scenarios and then explain the chosen scenario to be evaluated and the SIM application designed for that scenario. The implementation of the application and its simplifications will be discussed on the next chapter.

8.1. Future SIM aspects to improve trust

The first aspect that motivates the usage of the Future SIM as a trust platform is its characteristics as a secure module. The SIM, as a smart card, is equipped with cryptographic keys and a secure software and hardware environment. This security is important towards adding confidence to trust assessments based on the SIM. The SIM is capable of protecting the identities, offering mutual authentication between peers and ensuring confidentiality and integrity to the communication channel.

As mentioned in the previous chapter, the trust assessment can be based in the dispositional trust of the agent, the historic of interactions and feedback from 3rd party recommenders. The Future SIM and its firewalled application environment can protect the dispositional trust by protecting the software agent that implements this trust against hacking attempts. It can ensure that just the right application can access and update the interactions historic. Moreover it can authenticate and secure the exchange of feedbacks between the 3rd parties and the *trusting agent*.

The Future SIM greatly expands the connectivity mediums in comparison with the SIM cards massively deployed so far and other pervasive devices. It allows the SIM applications to communicate with other SIMs and platforms by SMS, Cell Broadcast, NFC and IP datagrams through IEEE 802.15.4-2003, IEEE 802.11 interfaces on the SIM or through BIP. Besides that, the SIM card could communicate through Bluetooth or Infrared via the mobile phone interface.

This vast range of communication channels position the future SIM as a great platform towards the Internet of Things, a scenario where all objects would communicate to each other. This would allow seamless exchange of services between objects and ad-hoc management of those. As mentioned in the background study, the SIM is already being used in M2M communication for fleet management and Point of Sales terminals. The works from ETSI towards the SIM in the M2M business, mentioned in the SIM chapter of this thesis, points that the future SIM will definitely not be restricted to mobile phones, but will be present everywhere. Thus, it will be able to serve as a trust component not only for people, but also for objects.

The SIM would work as the computational device for those objects, but also as their identity. As we mentioned before, the SIM is the ideal container for digital identities, not only for people but also for objects. By inserting a digital identity into the objects signed by a Trusted Authority, users and other objects would be able to connect to each other and to prove that they are communicating with the desired agent and that the communication has not been tampered. The federation between the Trusted Authorities and the identification factor in the SIM can tackle several cases of falsification and impersonations.

The identities can be represented by secured data and especially by keys and certificates. The SIM can assign different and protected security domains for those keys. By implementing the latest Global Platform specifications, keys can even be securely inserted after issuance. Moreover, in the standard smart card personalization systems, key can be done through very rigorous key management procedures complying with the needs of the most rigorous Trusted Authorities.

As a result, a species of Identity Selector just like the one mentioned in the industry identity standards such as Microsoft's Geneva and Higgins Project could be implemented on the Future SIM. The different security domains and access policies of the card would allow the selector to handle both non-critical identities (such as social network user account) and critical identities (such as the bank card or national E-Id). As we reviewed in chapter 6, several identities are already implemented in smart cards, and specifically in the SIM Cards. There are already some deployed cases such as the FINID and MyKad where more than one identity is implemented in the same card.

By managing and storing the user's identity, we eliminate the need of carrying multiple devices for identifying towards several systems and make use of Single Sign On. While the identification offers the possibility to personalize interfaces, services and communication modes for the user. For example, a user entering a job fair could select his professional identity and enable companies to retrieve his CV or Business Card seamlessly. Or when entering a music store, he could select an identity which have his musical taste as an attribute (such as a music-related OSN) and, based on that, receive music recommendations. In fact, this selection could be triggered automatically by location context adaption from the SIM card.

Some could be worried about privacy issues on the last case, but the amount of information shared by the user's identity should be over his control. It would be up to the user to decide if, in the music store, he would identify himself; and, if he does, he could choose to identify himself through a pseudonym or his real name. Moreover, he should be able to decide how much of his musical preferences he would share and if he would share it with all agents around, or just *trusted agents* based on their reputation or trust federation. It is important to keep in mind that there is a trade-off between the loss of privacy in revealing/sharing information and the gains in personalization of services and contact with other agents. The trust itself between agents is only acquired as they share information between themselves and the identity selection feature together with attribute sharing can help trust building.

Some inputs that can help in the identity management are the sensing capabilities of the Future SIM. As previously described in the chapter 5, there are deployments of location, motion, presence and time sensors in the SIM, besides the ones it can make use through the mobile or other device in which the future SIM is plugged. The insertion of the future SIM into the Internet-of-things further expands its sensing capability as it can receive context information from other objects. The Future SIM could use the sensed information to feed attributes of managed identities; and, as well, to gather other attributes information to help inferring context. For example, it could access the user's corporative agenda through his corporative identity in order to sense if he is busy or not. In the other hand it could transparently feed an OSN identity attribute with the real-time physical location of the user.

The sensing in the SIM card can enhance the trustworthiness of an attribute data related to the SIM card holder. The sensors' outputs correspond to real context evidences (in case of a physical sensing) towards the agent or a relationship it holds. It can prove that that a user is where he claims to be, that the environment is as it is sensed by temperature, humidity or other sensors. It can provide evidences that the user really bought a product that he is reviewing or that he really had a contact with another user in a sensed context. This is useful not only to enhance an already established trust relation but also to seamless build trust relations. It offers clues to challenges such as unbiased profiling the user and his relationships with other agents. Moreover the SIM is a tamperproof device and it serves as safe container or middleware for the sharing of those sensed evidences.

Still there are more reasons to have this context sensing in the SIM instead of other platforms. The SIM card is a ubiquitous device which most of the people are used to carry everywhere.

Most of those mentioned services and capabilities will be fully feasible on the future SIM also due to the new advances in the SIM Card as a computer platform. The introduction of the *JavaCard* 3.0 and the SCWS make it much easier to develop powerful applications in the SIM Card, making use of multithreading, application intercommunication and support for complex data structures. In the hardware part, there are also great prospects on the future SIM once the USB physical interface is being standardized and there are SIM prototypes with memory capacities of gigabytes.

8.2. Chosen Trust Scenario

As described in the section 8.1, some of the areas where the Future SIM can be used for adding trust to services are:

- Providing hard evidences toward agents (users or products).
- Triggering and controlling services; connection and attribute sharing based on context and on trust relationship between identities managed by the SIM.
- Creation, contextualization and trust attribution to ad-hoc relations between SIM agents.

In order to design and implement one of the mentioned cases, we needed to focus in an application case scenario where the Future SIM would be of great contribution.

8.2.1. The Process of Choosing the Scenario

In order to define which aspects of trust we would deal with and in which scenario we would develop our application, there were 2 possible approaches.

The first approach consisted in:

1. Establish the scenario (the “problem to be solved”).
2. Define the relevant parameters for the scenario.
3. Establish a trust model for the parameters and the scenario.
4. Design the application and evaluate it afterwards.

The second approach had the same steps as the first one, but with an order slightly different. There, we first define the trust model we would like to build, and based on that trust model, select input and output parameters. Then, we finally choose an application where this trust model and parameters could enhance its value. Afterwards, we would go the application design and evaluation.

We decided for the first approach, once we were more interested in the application cases of the future SIM than the trust modeling itself, it also seemed more natural to start from the application scenario.

8.2.2. Choosing the Scenario

We have identified three application scenarios that we would be interested to work with:

- Reputation systems: For ensuring the trustworthiness of people, opinions, products, companies, etc. In those we would endorse trust by sensing hard evidences and creation of trusted networks between the users in the system.
- Ride sharing scenario: Enabling people to share spaces on their vehicles during trips, regular commuting. There would be possibilities to enhance trust asserted by providing hard evidences and ad-hoc trust network creation.
- Food chain security: Tracking the product chain – specifically food- from the raw material (the cow, the farm, the fertilizer used for the growth of the pasture, etc) to the shipment to the supermarket and the storage conditions at the supermarket. The hard

evidences given by the SIM could protect the products from falsifications and illegal alterations.

We judged the 3 scenarios on how much we would be willing to work on it, on the business opportunities around them, and level of familiarity about the scenario. After voting for them, we opted for the first scenario.

The scenario of the reputation system was still quite broad as for example it was not defined in which agent (product, servicer or people we would be working with) nor on which aspect of trust we would work with. Therefore, we again narrowed down to 3 possible cases.

- Product Review for buyers: Such as reviews of computer models, digital camera models.
- Provider Reviews: Reviews of service or product providers such as hotels, bars, shops etc.
- People Review: Review on people for application of house sharing, as offering your place for someone to stay at. Examples: Hospitalityclub[66], Couchsurfing[67].

We foresee a bigger business opportunity and application areas in the 2 first cases and decided to detail trust related use cases for them. For both the product review and the service provider review the trust cases are similar, but a little bit broader for the service provider/company because we have both the reputation of it based on its characteristics and based on its services or products. The trust enhancement cases we have detected were:

- Automatic recommendations could be inferred from the fact that someone has bought a product several times. For short life consumption products such as chocolate, shampoo, etc), it denotes that the buyer support the product.
- Assert credibility to recommenders based on hard-evidence towards the usage or acquisition of products. The opinion of people that have never used the product may not be as relevant as a frequent user or a one-time user.
- Assert credibility to recommenders based on his identity inherited profiles. The credibility of the reviewer is also dependent to personal and business interests. A review from someone from HP about an HP product may be tendentious. In the other

[66] http://www.hospitalityclub.org/
[67] http://www.couchsurfing.org/

hand, the user may not rely on the credibility of a review due to different personal interests (such as the reviewer does not like the same music style as the user).

- Creating a personal network of trust so the credibility of the review can be inferred from the personal relationship between the user agents.

At the same time, we randomly took the example of a fish shop in order to find quality assessment criteria for its review and we came up with the following ones:

- The shop's profile: quality standards (a shop known for special quality fish, fresher fish), price standards (a shop with reputation of having cheap fish), environment (if the shop is dirty or clean), product variety, friendliness of the staff, etc.
- The location of the shop can be meaningful for the user as if it is easy/hard, far/close to get there, and also it contributes to the profile (if it is in a rich/dirty/cozy neighborhood).
- For the fish itself we have some hard evidenced characteristics such as if the fish is fresh, how old is it, where it comes from. But we also have some relative characteristics such as if the fish tastes good, is easy to prepare, etc.

At the end of the discussion around the quality assessment criteria and trust cases, we decided to focus on creating a seamless trust between users so based on that trust they can ask themselves and consider the reviews.

We noticed that one of the biggest challenges is on defining and modeling the credibility of the other agent, and that the mechanisms of relationship building of the current relationship networks (the best starting point for attributing trustworthiness relations between users) could be greatly enhanced.

As mentioned in the Trust Chapter, often the relationship building in OSNs merely consist of a request that is accepted despite the fact that the two users to do not really hold a relationship. Also, the lack of a seamless method for creating those requests avoids that people who maintains a real relationship can digitalize them on an OSN.

Based on that problem, we decided to work on a passive relationship builder. This passive relationship builder will sense context information during real physical relations between the users and use this sensed data to add trust information to OSN. It is a trust case that can greatly benefit from the pervasive and context sensing characteristics of the Future SIM.

8.2.3. Defining the relevant parameters

As possible input parameters for this passive relationship builder we have identified:

- Identification. The simplest use of the identification attribute is to use it for symbolizing the users in the passive relationship. In other words, we use the digital identities of the future-SIMs owned and carried by the people engaged in the physical relation in order to determine who is participating on the relation.

  A more advanced case is to use attributes linked to the several identities managed by the future-SIM. Gathering all those attributes (such as the home address of the user, his profession, interests, family tree, etc) can offer almost unlimited possible inputs to enrich the description and contextualization of his profile or a trust relation held with other users. It would be possible to infer if the users share common interests; have distinct or similar points of view; are family related; etc.

  Despite this rich potential of those attributes information, we decided not to explore it once most of the work would be related to data mining and crawling. Nevertheless, we expect it to be possible to use those in the future due to the widespread of Identity Attribute Providers and the opening of API's to access OSN content.

- Proximity and Location. Proximity and location are rich contexts and the most explored ones by the industry. We can infer the level of acquaintance between two people based on how much time they spend together. In addition, more context information, such as the environment where they are, can bring further accuracy to this relationship inference. For example if the users are close by in the bus is less likely that they know each other than if they are at the same working office. Although for the first case, if this situation repeats often, we can create a seamless relation in the context that they are common commuters, take the same bus at the same periods.

The future-SIM and its' native radio interfaces such as NFC, IEEE 802.15.4-2003 and IEEE 802.11 can sense other future-SIMs in the radio range, as a proximity sensor. Thus, it is possible to shape a relationship based on the identity of the radio interface or trigger the exchange of the digital identities between users. Another mean to do the same would be to compare the physical location retrieved by location sensors and compare with the real-time location attributes of other users on the OSN in order to define if they are close to each other.

The location itself can be either relative (at shopping mall, home, office at work, meeting room at client's office) or absolute (in geographical coordinates, city, address, etc). Nevertheless, the absolute location could be translated to a relative location information and vice-versa with the mapping of points of interests, which have been happening quite often thanks to the geo-location platform offered by Google Maps[68]. The location data can be used to help characterize the relationship, such as mentioned in the case of the bus commuters. The position can be gathered by the Future SIM card through direct location sensors such as GPS, CellId or indirect ones such as WLAN hotspots positioning.

- Motion: The motion can offer some context information towards the environment or activity of the user, helping the inference of the relationship context. If a person is moving and dislocating, the speed can tell in which vehicle he is (if he is in a vehicle). Or we can sense his ergonomic position (such as if he is stand, sit, laid, etc). The ergonomic position can give context information about the state of the user as if he is idle or moving, or it could be combined with other context (such as location) to infer his activity.

  At last, even position could be inferred from motion based by registering the starting point of the user and tracking his movements. As mentioned in the SIM Card chapter, the SIM card industry has already announced SIMs with built-in accelerometer which would make it possible for the future SIM to sense the motion.

[68] http://maps.google.com/

- Soft sensing: Those are sensors based on inference of the user actions within his devices. Based on the use and software running on his mobile we can find out if he is idle or busy; if he does not want to be disturbed (with the mobile in silent mode); or even his precise activity (engaged in a phone call, playing solitaire, etc). The interfaces between the SIM and the mobile such as the SIM Toolkit, SATSA and SCWS can act as the bridge for this soft sensing to reach the Future SIM.

Due to the limitations of hardware available and the time frame for doing the Master Thesis, we have not used all those sense parameters. But we notice that the aggregation of all of them could greatly improve the efficiency of the seamless relationship builder. Anyway, we will describe in details our implementation in the next chapter.

8.2.4. The Trust Model

As described in the previous chapters, the broad range of sensors to be expected in the Future SIM and the digital identities stored there can offer a comprehensive amount of inputs for creating and maintaining direct trust relations between the users.

For defining the trust value for the seamless trust relations we can consider the dispositional trust, the historic-based trust and recommender's trust.

The dispositional trust can be modeled by varying the weights given to each assessment criteria, distribution of fuzzy values or weight factors in the deterministic equations for calculating the trustworthiness value. This is a complex theme that some attribute information retrieved by the users' identities could help.

The multiple identity capability of the Future SIM could be explored in order to retrieve recommendations concerning a new seamless relation. If some of the identities of the *trusting agent* are part of an OSN or a contact network, the degree of distance towards the *trusted agent* could be fetched; or he could find contacts on his trust network that could work as *recommendation agents* towards this new seamless relation. It may represent a great potential in relation to trust transitiveness. However we will not consider transitiveness inputs in our implementation due to the fact that we want to focus the practical part on the sensing capabilities of the SIM. Additionally, transitiveness experiments, such as the one done in the PhD Thesis (Ann Golbeck, 2005), involves a lot of time and people, making them infeasible for this Master Thesis.

One area where the Future SIM sensors can provide a great, and so far unexplored, contribution to trust is to seamless capture hard evidences of the day-to-day relationship interactions between agents. The captured data can generate trustworthiness values to module dispositional trust or the final trust based on the historic of those direct interactions for each context. As mentioned before the location sensing capability, possibly helped by the motion and soft sensing, can fetch context and proximity. Then, it is possible to use those to assert the level of trust in the relationship.

A complete approach for calculating the trustworthiness value should make use of the soft sensing capabilities mentioned on the previous section in order to be able to use the activity context of the agent as an input. It should also make use of the great amount of information that can be retrieved through the identity attributes. However, for this Master Thesis we will limit the input to the location context and the direct contact information (instant and duration) for calculating the trustworthiness value.

9. Implementation

Although most of the work on the Thesis was on the assessment of the future SIM capabilities, on the possibilities to endorse trust in related services, we decided to develop a small prototype of the described seamless relationship builder.

Besides hardware limitations, as the ideal platform is the Future SIM which does not exist yet, we have a time limitation for the thesis. Therefore, we focused our implementation on building a proof-of-concept showing that it is possible to deploy the mentioned seamless trust builder on a Future SIM.

9.1. Implementation Platform

We initially considered the following application formats as a substitute for the Future SIM in our application development: a *JavaCard* Application, a J2ME application, a Sun Small Programmable Object Technology (Sun SPOT) application and a J2SE application.

As we were more interested on the networking and sensing capabilities of the Future SIM, we would need to deploy the application into a wireless device connected to some of the sensors that are being embedded to the most advanced SIM cards. Given that condition, a J2SE application running in a standard computer was discarded from our options. Moreover the J2ME handsets and *JavaCard* SIM card we had in the Lab did not have many sensing support as the Sun SPOT device. Thus, we have chosen to implement the seamless trust builder in the Sun SPOT.

9.1.1. The Sunspot Platform

The Sun SPOTs are small, wireless devices embedded with a few LEDs, 3 different sensors (temperature, light and an accelerometer) besides I/O general purposes pins that enable the connection of other sensors (Sun Microsystems, 2009). Instead of having an operational system, the Sunspot host a small-footprint J2ME CLDC 1.1 java virtual machine called Squawk running directly over the hardware. The underlying hardware consists of a 180 MHz

32 bit ARM920T core processor with 512K RAM, 4MB of Flash Memory, a 2.4 GHz IEEE 802.15.4 antenna and a 2G/6G 3-axis accelerometer (Sun Microsystems, 2009).

The wireless 802.15.4 network transceiver is built on the TI CC2420 chip and it supports received signal strength indication with 100dB sensitivity. It transmits output power setting from 24dBm to 0dBm. The RSSI values range from the strongest value of +60 to -60 (Zennaro, Ntareme, & Bagula, 2008).

In Table 4, we enumerate the advantages and disadvantages of implementing in the Sun SPOT:

Table 4: Advantages and Disadvantages of Sun SPOT Implementation

Advantages	***Disadvantages***
Virtual Machine and API similar to *JavaCard* 3.0	Somehow infant product (not much documentation available)
Accelerometer Sensor integrated	No GUI on the device itself
802.15.4 radio communication interface	No absolute location sensor
RSSI radio feedback can sense proximity	Limited cryptographic support
Location context through location packets from 802.15.4 hotspot	
Portable and Mobile	

While the Future SIM would be able to possibly capture location with GPS sensors, CellId information and radio hotspots, in our case we are restricted to the location inference through 802.15.4 hotspot mapping. The WPAN range of the 802.15.4 is very limited. In accord to testimonies from Sun SPOT developers at the Sunspot Forum[69] and tests done in (Zennaro, Ntareme, & Bagula, 2008), the ranges are from around 10 meters in closed environments and up to 50 or 100 meters in open environment with few interferences. The accuracy of the RSSI positioning with the 802.15.4 is also a polemic topic. While (O'Dell, O'Dell, Wattenhofer, & Wattenhofer, 2005) shows that this may not be that accurate in a more real environment with

[69] http://www.sunspotworld.com/forums/

several nodes, (Lowton, Brown, & Finney, 2006) presents that accuracy of around 10cm can be achieved through RSSI metrics.

The Sun SPOT is equipped with the LIS3L02AQ accelerometer which can measure the acceleration over a scale of ± 2g or ± 6g. It can be used to measure the motion of the SPOT or its orientation with respect to the gravity. The accelerometer API offers mechanisms to get the values from the 3 axis, to get differences in relation to a threshold, the acceleration vector intensity and finally the spot orientation (Goldman, 2007).

Theoretically, we could use the accelerometer to get an idea of the user activity, his postural position and even to calculate his position. For example, a study done by (Karantonis, Narayanan, Mathie, Lovell, & Celler, 2006) tries to identify some user activities with focus on elderly monitoring. They try to recognize situations such as if the user is standing up, sitting, falling and walking. They also used a device with small memory (in their case 2KB of RAM). As a final result, they managed to detect with good accuracy the changes between activity and rest and some other patterns, but the information if the person was in fact walking was not accurate.

One point that must definitely be considered for a less experimental prototype is the battery consumption. It will be important to wisely use the sensors, LEDs and radio once they greatly impact the power. In the specific case of the Sun SPOT, it is estimated a battery life time of 7 hours with a busy CPU and radio (Sun Labs, 2008).

9.2. Application Design

The seamless relationship builders will work by sensing each other by the exchange of proximity broadcast datagrams. Once they sense each other, the one that received the broadcast will start mutual authentication handshake, such as TLS using X.509 certificates. After the sharing of a secret through TLS, they will exchange a pseudonym which will be used to identify the user in the trust relationship (it is worthy to mention that the X.509 certificate itself could act as this identifier). Afterwards, they will keep the connection alive for as long as they are nearby, in each other’s radio range. After the connection is dropped, the entities will keep the information of the other's identity instant of contact, location and duration where the contact happened. This information will be stored and the current trustworthiness between them will be calculated for instant of time, based the historic of interactions.

9.2.1. Sensing

The location information will be retrieved from location broadcast datagrams emitted by a hotspot acting as a location information provider. In order to do so, we developed this small application in the hotspot that periodically broadcast its location and configured the location for each hotspot used. This application is to emulate the fact that the Future SIM card would be able to retrieve the location from a mapped CellId, GPS or even a similar hotspot. The location packets will provide a contextual location that in our case was divided in {home, outdoor, my office and office's corridor}. Nevertheless, the real application should handle more types of contextual locations and possibly be able to convert absolute geographical locations into contextual ones.

Initially, we thought about recording the accelerometer data during the interaction in order to infer which kind of activity the user was engaged into. However, experiments we have done using the accelerometer with the Sun SPOT Telemetry Demo (which captures and plots accelerometer data) revealed several difficulties in categorizing the activities. We tried to observe the values of the "X","Y","Z" and the total intensity for situations such as walking, running, sited, laying down and having the sunspot in a pocket, inside a bag and being held close to the ear as a telephone. Depending on the orientation of the spot and the type of physical contact applied to it, the patterns would change. In order to be able to sense the user activity, some type of auto-calibration should be developed to adapt for the different ways of carrying the Sun SPOT.

Due to the time limitation of the project, we decided to use the accelerometer just to sense if the user is moving or if he is stopped. We use this information as we designed that only the SPOTs moving will send the proximity broadcasts. By that, we have less chances of having channel congestion and we save some battery. This is important because, as noticed back in section 5.3.1.1, a positioning sensor can consume a lot of battery. As always one SPOT will have to move in order to get into another's range, this will not exclude any interaction.

For sensing the proximity, we periodically exchange the RSSI measurements with the keep-alive packets until the connection is broken due to the fact that one of the SPOTs leaves the radio range. Although the radiation power is proportional to the distance, it suffers great variations due to interferences. The presence of objects, people and other radio waves diffract and bounce the 802.15.4 radio waves. As a result, some tests we have performed have signalized more less the same RSSI for spots in the same rooms or in rooms separated by wall. Because of that, we decided to contextualize the proximity in our prototype as {reachable, not reachable}. However, in some situations it may be possible to distinguish the contact between the users as if they are in front of each other, close by or in different rooms.

9.2.2. Identity

The Sun SPOTs have a default repository of trusted certificates. In fact, they just allow the installation of applications that have been signed by a private key corresponding to one of its trusted public keys (Sun Microsystems, 2008). However, those operations are carried transparently and there is no native API to create its own key pair, sign data, etc.

There is a non-native library called spots-security[70] which is supposed to create a key pair for the spot and sign the key with the SDK key. The library also configures the SPOTs to trust keys signed by the same SDK key and offers TLS (with server authentication only) support for the radiostream protocol, a Sun SPOT protocol to transmit stream of radio packets. Since it does not offer the possibility to change those root keys or add other key pairs, we need to use those for the TLS handshake. In the real case the key pair could be chosen (possibly based on the current context) by an identity selector in the Future SIM; and it should be used mutual authentication.

Unfortunately by the time we have finished this thesis the spots-security was still not working perfectly. The key pair was being generated correctly as we could see by listing the Sun SPOT trusted keys with the command “listtrustedkeys”, but the TLS handshake was not working. A similar problem with the TLS handshake has been posted on the Sun SPOT Forum without a reply and the developer of the library could not manage to help us in time. Due to this

[70] https://spots-security.dev.java.net/source/browse/spots-security/

problem, we ended up using a non-encrypted handshake in the application and the MAC address as the spots identities, although a TLS one should be done on a real prototype.

9.2.3. The Implemented Application

In order to physically emulate our scenario we have used a fixed Sun SPOTs basestation as contextual location provider. The basestation is also responsible of being the communication link between the agent SPOTs and the database server. In what concerns the location providing, the basestation just broadcasts IEEE 802.15.4-2003 packets in a specific port. Those packets are tagged as “location information” packets and contain a byte data that representing its location as “home”, “office”, “corridor” or “outdoor”. The basestation’s locations are configured directly on the source code before execution in order to set different locations in the experiments.

On the agent application we have a few concurrent threads running in order to broadcast and sense proximity and sensing the movement. Besides that, once another agent is sensed, a new thread is started to establish a communication between agents. Once the communication is over, the thread will send a message to the main application with the data about the interaction. And the main application will send the data to the database. A sketch of the roles and communication of the spots can be seen in Figure 21.

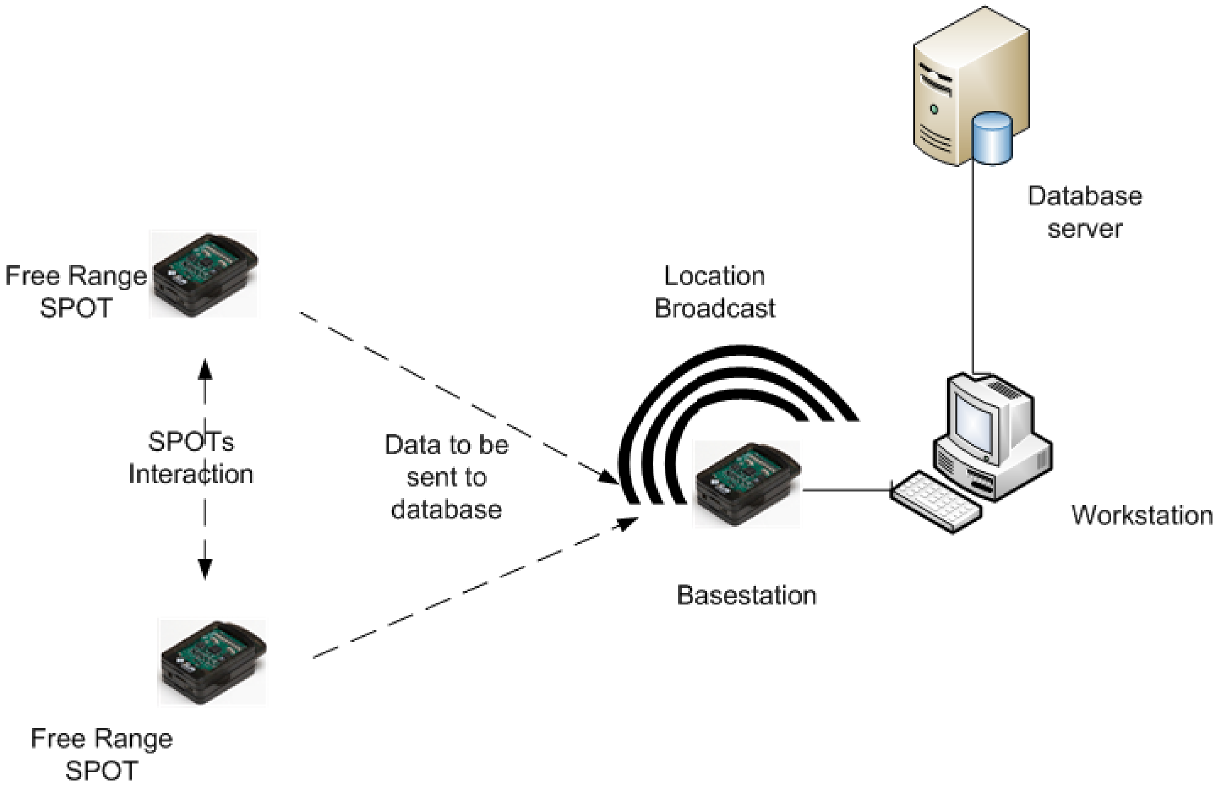


Figure 21: Trust Builder Diagram

In the next section, we describe all the threads and present a SDL diagram of the communication of the threads. We have chosen to model them in SDL, due to the concurrency and real-time characteristics of SDL which matches with the threaded and multi-agent architecture we used in the application.

9.2.3.1. Application threads description

First we start by describing the main process thread. Its first action is to start the accelerometer and broadcaster threads. Then, it mainly acts handling the sensors' and threads' inputs by:

- receiving location information and updating the location variable;
- receiving proximity information and starting the handshake process, in case the other agent does not have a handshake process established with this one. The track of the active communication is done by keeping a table in the memory with the list of target MAC addresses enrolled in active communications;

- receiving the interaction information (duration and other agent data) from the communication handshake thread, calculating the trust values of the interaction and forwarding it to the basestation so it can add the data in the database server;
- receiving the accelerometer data from its thread and starting or stopping the proximity broadcaster thread;

The picture below represents the messages exchanged by the Main Thread with the other threads. More details about its logic can be found in its source code in the "StartApplication.java" file.

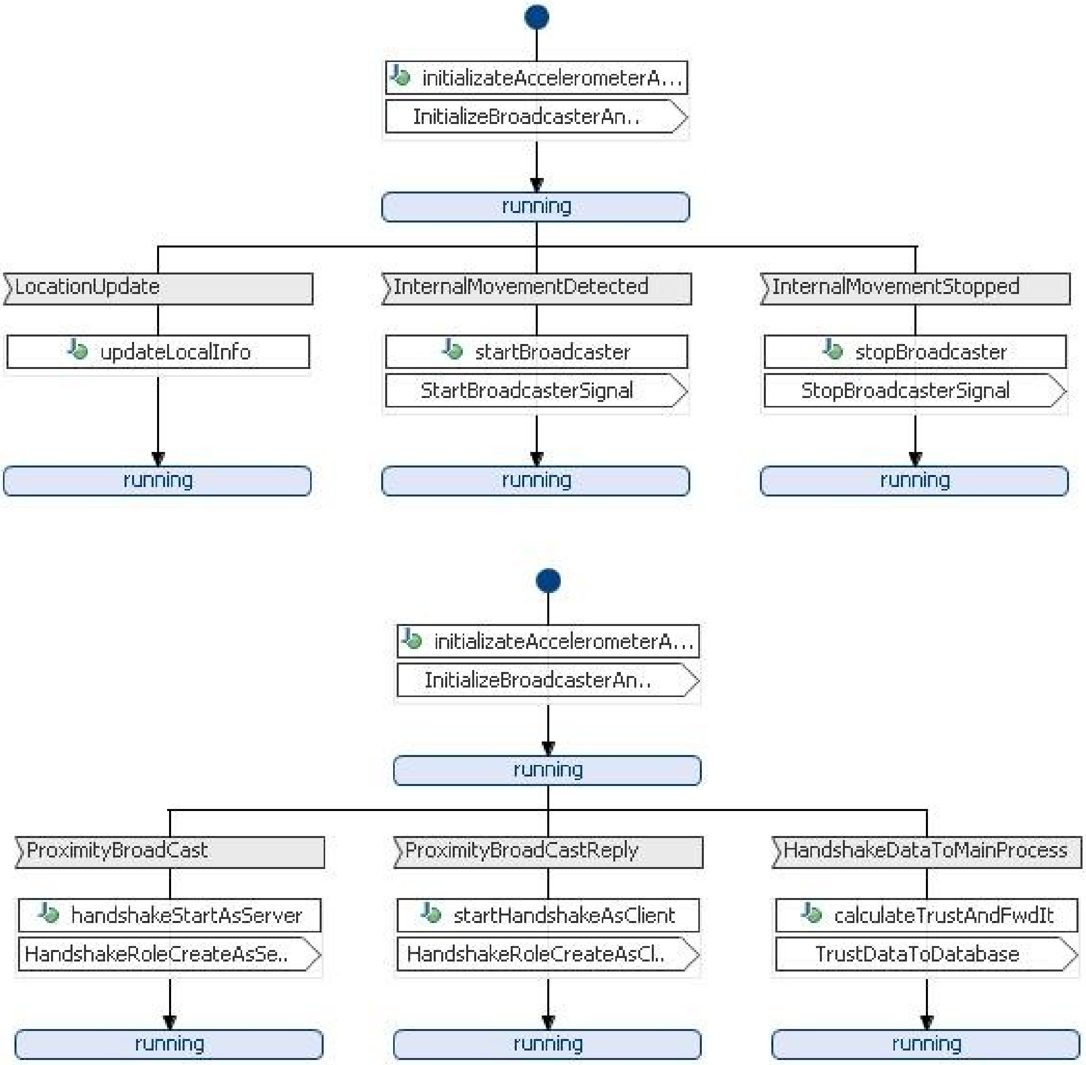


Figure 22: Main Process Diagram

The Accelerometer Thread, described in Figure 23, will listen for acceleration changes and dispatch the motion event (existence or absence of motion) to the main thread. By that way, we encapsulate the motion detection logic and we get flexibility to tune the motion detection algorithm without affecting the main application.

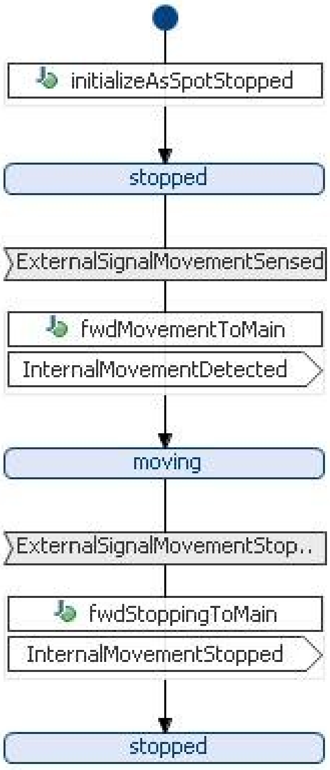


Figure 23: Accelerometer Thread SDL Diagram

The logic used for detecting the movement was to get initial acceleration intensity, and, whenever this intensity varies more than a fixed threshold for at least a fixed amount of time (samplings) it is considered in movement. When the intensity is back to the initial value, during the same number of minimum samplings, it is considered stopped. More details about it can be seen at its source code in the file “AccelMonitorTest.java”.

The Proximity Broadcast Thread will broadcast proximity tagged packets periodically until the thread is stopped by the main process. The purpose to have it as a thread is to remove this task from the main application and so it can be run in parallel despite the main process. Its code can be found in the file “Broadcaster.java” and its SDL diagram in Figure 24.

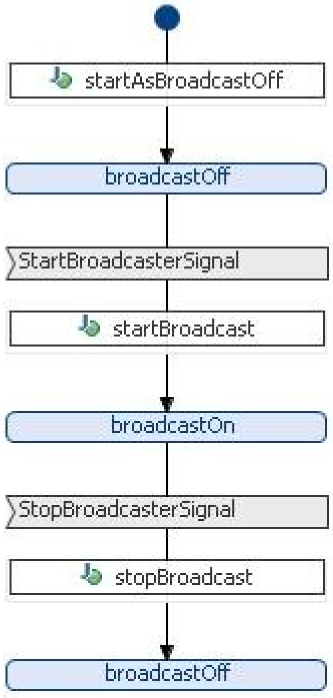


Figure 24: Proximity Broadcast Thread Diagram

Once the main thread receives a sensing broadcast packet, it will start a handshake Thread as the server for the interaction monitoring between the agents. The newly created handshake server thread will reply to the other spot a “Proximity Broadcast Reply”. By that, the main process in the other SPOT will start a client handshake thread.

Both handshake threads will monitor the connection by exchanging application ACKs, and once there is no answer, it is assumed a disconnection and both threads will submit the interaction data (MAC of the other peer and duration) to the main process and the handshake threads will be terminated. The SDL diagram for the handshake treads can be seen at Figure 25, and its source code is in the “SocialHandshake.java” file.

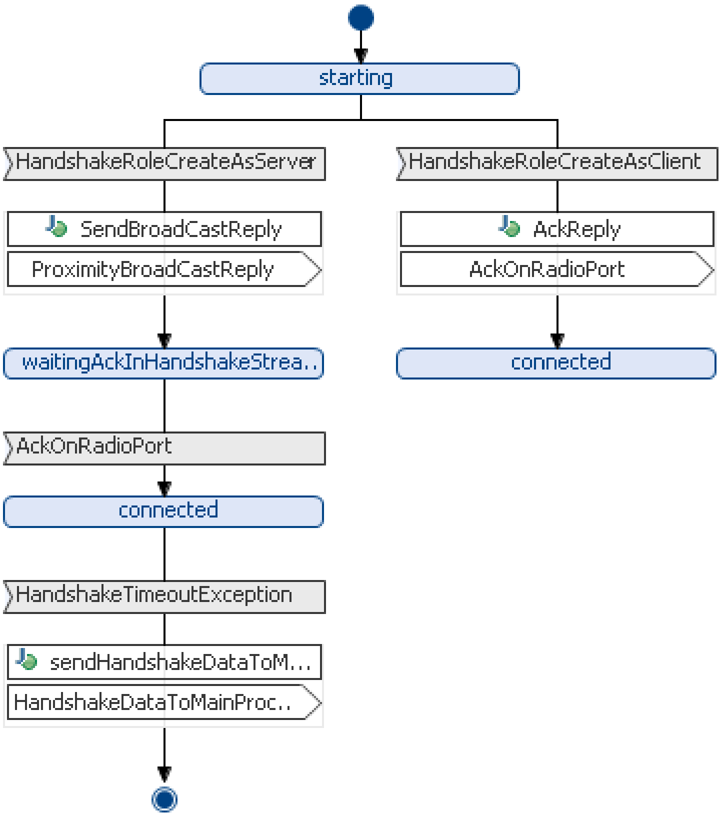

starting
HandshakeRoleCreateAsServer
HandshakeRoleCreateAsClient
SendBroadCastReply
ProximityBroadCastReply
AckReply
AckOnRadioPort
waitingAckInHandshakeStrea..
connected
AckOnRadioPort
connected
HandshakeTimeoutException
sendHandshakeDataToM...
HandshakeDataToMainProc..

Figure 25: Handshake Thread SDL Diagram

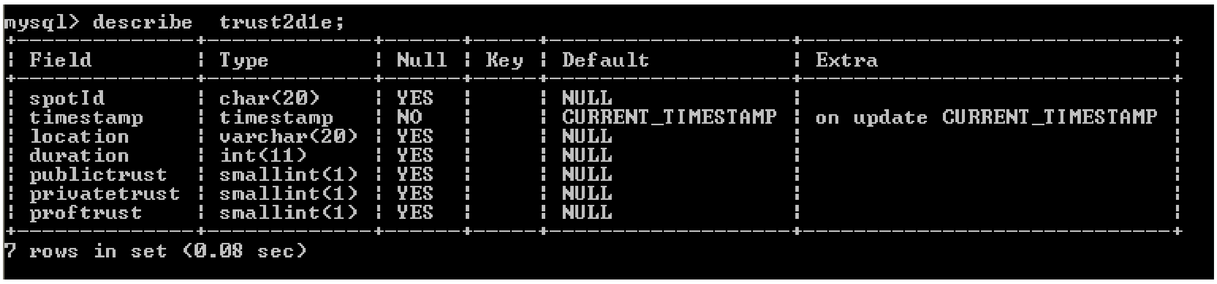
In fact, we tested and coded some other context sensing capabilities in the Handshake Threads, although we did not use them in the final application. One of them was the measuring of the transmission power through the RSSI. We wanted to define some categories of proximity and use them to enhance the trust model, but the measurements were not accurate enough to do it, as mentioned in the section 9.2.1.

Another test was to attribute a second identity, an alias from the existing social network Last.fm, to the SPOTS and retrieve contextual information from it. We successfully manage to retrieve the favorite artists of the users and their "musical compatibility" by calling the Last.fm REST API through HTTP request from the SPOTs and routed by the basestation. The source code for that last capability can be found on the file "LastFMConnector.java".

We have not used the context retrieved the Last.fm social network as we decided to focus on the physical sensing. However, the experiments showed that it can be quite simple to retrieve this kind of information from the Social Networks that provides some open SOA API. This could be used for a model that considers this soft context through the Sun SPOTs platform.

Although it was possible to deploy the database in the sunspots, we decided to do it on a MySQL server running in the machine attached to the basestation. This decision was mainly motivated by the fact that the SPOTs do not have a native display and it would be easier to manage a central database for the experiments.

The database contains one table for each free-range spot *trusting agent*. In that table we list the following information about the interactions between the spots: ID of the other spot (*trusted agent*), timestamp of the interaction, location, duration and the values of public, private and professional trust assigned to that interaction, as shown in Figure 26.



```
mysql> describe  trust2d1e;
+--------------+--------------+------+-----+-------------------+-----------------------------+
| Field        | Type         | Null | Key | Default           | Extra                       |
+--------------+--------------+------+-----+-------------------+-----------------------------+
| spotId       | char(20)     | YES  |     | NULL              |                             |
| timestamp    | timestamp    | NO   |     | CURRENT_TIMESTAMP | on update CURRENT_TIMESTAMP |
| location     | varchar(20)  | YES  |     | NULL              |                             |
| duration     | int(11)      | YES  |     | NULL              |                             |
| publictrust  | smallint(1)  | YES  |     | NULL              |                             |
| privatetrust | smallint(1)  | YES  |     | NULL              |                             |
| proftrust    | smallint(1)  | YES  |     | NULL              |                             |
+--------------+--------------+------+-----+-------------------+-----------------------------+
7 rows in set (0.08 sec)
```

Figure 26: Trust Database

Although the SPOTs are performing the calculation of the trust values, this could be done directly in the database, saving some processing power from the SPOTs. In case that there are many SPOTS and many interactions, another possibility would be that each *trusting agent* would have a trust database towards each *trusted agent*.

The instantaneous trust calculation was also done with another Java application running in a PC, its source code is on the file “Main.java”. This could as well have been implemented in the SPOTs, and again, the reason for not doing so was to make it easier to read the results and evaluate them. In the case of a real application on a Future-SIM, it could be uses the mobile display for that.

9.2.4. Trust Inference

The fuzziness aspects of trust mentioned in the Trust Chapter makes it a challenge to design a model which can accurately generate realistic trustworthiness value to the trust relations. In order to reach those, it is necessary a co-work with professionals of sociology and psychology field. Besides that, it needs the realization of experiments with a large sample of unbiased social interactions involving a lot of time and people. Moreover we have not found any trust model in the computational trust related literature that tries to infer trust based on the physical relation between the agents. Due to those difficulties and the limited scope of this thesis, we will design here an inference trust model that most likely is far from a perfect trust inference, but that can be used as a starting point for further development.

As mentioned before, the Sun SPOT sensor will sense location as a contextual location whose value ranges from:

- Home: Which represents the place (apartment, house or similar) where the user lives.
- My office: Which represents the working place of the user (his room/cubiculum at his company, his classroom).
- Office's corridor: Which represents a passage place inside the institution where the user works (such as the corridor, entrance hall or coffee machine area).
- Outdoor: Represents any public place such as a park, a shopping mall, streets of a city, etc.

We are aware that there are many more different contextual locations that could be described, as there are different ways to model them. However, for the sake of simplicity we will only consider the ones mentioned above.

The duration of the contact will act as another input in our system. In order to have more precision, different duration ranges should be defined for each location context. In a more realistic scenario, a contact at home characterized as short will have a different duration then a short contact at the office. Nevertheless, we will use a single range categorization as defined in Table 5.

Table 5: Duration Quantization

Duration Range	***Context***
Less than 30 seconds	Equivalent to the situation where the users just pass by each other (PASSBY)
From 30 seconds to 10 minutes	Equivalent to a Short/quick contact, such as stop and greet each other; (SHORT)
From 10 minutes to 4 hours	Equivalent as being engaged together in an activity (MEDIUM)
More than 4 hours	Equivalent as sharing a daily environment, as a work/family/school mate (LARGE)

Our trustworthiness value will be constructed by the result of a deterministic equation. We have decided for that approach because it is the most simple to model, and as we have mentioned, it is out-of-the-scope of this project to create a complex trust model. However, we believe that trust context is a very important factor. Thus, we designed our model as having the result of our trust calculation as a trustworthiness value for each of the following trust contexts: Private, Public and Professional. The private context symbolizes relationships on the private level (such as family and friends). The professional context represents relationship related to the person's profession or work. And finally, the public trust context tries to represent people that the user knows but that he does not consider as a close friend (for example, they have never been to each other's home).

On our model we are only taking into account the degree of familiarity or acquaintance between the agents, estimated from the physical contact duration. Both the willingness and capabilities are not being considered, once they would need a more complex sensing (based on degree of direct social network distance, global reputation, profiling or others). Being that said, our trust scale will be graded from 0 to 5, where 0 does not mean mistrust, but that the *trusted agent* is not known by the *trusting agent*. The Table 6 attempts to attribute some meaning, some equivalence, to the mentioned trust ranges in each one of the contexts.

Table 6: Trust Scale

Trust Value	Professional	Private	Public
1	They are from the same company	They are somehow privately related	Have met in public contexts
2	They know each other in the work environment	Acquaintance	Know each other from public activities
3	Work in the same department	Close Friend	Share a common interest
4-5	Work in the same room	Family	Are part of a common interest group or organization

On the matrix below we describe the trustworthiness value inferred from each contact. This value will be deduced from the context pair location and duration. We describe what we call equivalent situation as the type of situations that motivated us to assign the trust values based on the (Location, Duration) pair. We are aware that there are more situations to be taken into account, but we consider those as a starting point for a research in the topic. A longer study on the area should refine the values we estimated, create different customized duration intervals for each location and take into account the user profile (as some users may spend more or less time with other people).

Table 7: Trust Matrix

Location	***Duration***	***Equivalent situation***	***Profes. Trust***	***Public Trust***	***Private Trust***
Home	PASSBY	Someone passes by, no spoken contact	0	1	0
	SHORT	Possibly a home delivery or small talk at the door	0	1	1
	MEDIUM	Someone that is invited to come in;	0	2	3
	LARGE	Other people living at home or passing the night;	0	5	5
My Office	PASSBY	Someone passes by, no spoken contact	1	0	0
	SHORT	Asking for work-related information, small chat	2	1	0
	MEDIUM	Meeting	4	2	0
	LARGE	Working together in the same room	5	3	2
Office's Corridor	PASSBY	Cross each other, no spoken contact	1	0	0
	SHORT	Stopping and chatting in the corridor	1	2	0
	MEDIUM	Activity being performed as having lunch together	3	3	2
	LARGE	NOT DEFINED	-	-	-
Outdoor	PASSBY	Cross each other, no spoken contact	0	0	0
	SHORT	Crossing each other leading to small chat	0	1	0
	MEDIUM	Enrolled in an activity together: cinema, shopping, sport,etc	0	2	1
	LARGE	Enrolled in a daily activity together: travelling, camping, etc	0	4	3

We will store in a database all the three resulting trust values together with the instant of the contact, duration and location. In order to model the time degradation of trust, the instantaneous trust will be calculated as: a weight of 70% over average of the interactions considered as recent, plus a weight of 25% over the average trust from the interactions considered as past and a 5% weight over the average trust from the ones considered old. Some trust literature, as (Chang, Hussain, & Dillon, 2006), uses a logarithm degradation function and continue input values of time. In our case, we opted for a discontinuous approach of mapping time into categories, since it is easier to model and to update the values.

The definition of the categories "recent", "past" and "old" will be as specified in the table below. Just like for the trustworthiness values, those categories should be adapted for different cases and possibly extended.

Table 8: Time Degradation Categories

Category	***Time Frame***
Recent	Less than 5 weeks old
Past	Between 5 weeks old and a year old
Old	Older than a year

Summarizing, for each connection (*trusted agent*) and each contextual trust (professional trust, public trust and private trust) we would have the following formula:

$$T_{context} = \frac{0.7}{n_{recent}} \times \left(\sum_{i=1}^{n_{recent}} T_{recent_i} \right) + \frac{0.25}{n_{past}} \times \left(\sum_{i=1}^{n_{past}} T_{past_i} \right) + \frac{0.05}{n_{old}} \times \left(\sum_{i=1}^{n_{old}} T_{old_i} \right). (1)$$

Where "n" represents the number of interactions for the time frame category and " $T_{category_i}$" represents each one of the trust values of that time frame category.

Since we incorporate the time degradation factor of trust in our model, we store the interactions values on our database and we calculate the instant trustworthiness value in the moment that the trustworthiness is assed. For the storage we do it in a centralized database in a computer. This makes it easier to forward the trust to other agents but move the security concerns to a single point, which may become a single point of failure. In a real deployment, the Future SIM could be a better recipient for this trust database as it would be more user centric and its security would be assured by the SIM card security itself.

10. Experimentation

The experimentation in this thesis in fact started with the familiarization with the Sun SPOTs sensors and by evaluating, through tests which sensors could be used in practice for our seamless relationship builder.

As mentioned in the chapter 9, after a few tests with the accelerometer and the RSSI values, we decided to use the accelerometer just as a trigger for the proximity broadcasts and we did not use the RSSI values, as they were not offering information enough to successfully distinguish proximity characteristics in heterogeneous environments.

For the experiments, we had limitations on devices (just 3 sunspots available), time and people available for a behavioral simulation. Consequently, we decided to focus on generating the application code in the sun SPOTS and trying to generate trust values out of a few simulated situations inside the lab.

We created a small fictional scenario where we emulate a relationship between two users. In order to do so, we created a script describing their relationship, and the interactions that would result of their relationship. Based on that, we decided to individually test the trust model, and then, test the sensing acquisition and input of data in the database. It is a valid approach as we are doing an emulation of the scenario instead of the scenario itself. Moreover, it gave more flexibility to change values in the trust table (Table 7) and to try a different trust calculation. At last, by implementing the acquisition of the data for the whole emulated scenario with the sunspots would require too much time, due to the need of taking in consideration the duration and distance in time of experiments. We will further describe both experiments in this chapter.

10.1. Emulated Scenario

We consider as our emulation scenario the relationship of two friends that work at the same company but at different departments. Since they work at different departments, they have distinct routines. At work, they mainly meet once a week for having lunch and twice a week while having a cafe and a small chat at one’s office room. Their friendship goes beyond the office as once a month they either do an outdoor activity (such as going to the cinema, jogging, etc) or they have dinner at each other's house.

We will calculate their trust on those premises and, then, insert new events such as a trip together for 3 days, one of them going in vacation for a month and finally, one of them moving abroad for more than a year. Then, we will observe how their mutual trust is affected by those events.

The summary of the interactions based on the mentioned routine of 3 weekly meetings, a monthly meeting outside the company and the special events such as the vacation have been modeled as described in this table:

Table 9: Scenario Interactions

Week Nb.	Contact Nb.	Context	Location	Duration
1	1	Lunch	CORRIDOR	MEDIUM
1	2	Coffee	OFFICE	SHORT
1	3	Coffee	OFFICE	SHORT
2	4	Lunch	CORRIDOR	MEDIUM
2	5	Coffee	OFFICE	SHORT
2	6	Coffee	OFFICE	SHORT
3	7	Lunch	CORRIDOR	MEDIUM
3	8	Coffee	OFFICE	SHORT
3	9	Coffee	OFFICE	SHORT
3	10	Outdoor Activity	OUTDOOR	LARGE
4	11	Lunch	CORRIDOR	MEDIUM
4	12	Coffee	OFFICE	SHORT
4	13	Coffee	OFFICE	SHORT
5	14	Lunch	CORRIDOR	MEDIUM
5	15	Coffee	OFFICE	SHORT
5	16	Coffee	OFFICE	SHORT

6	17	Lunch	CORRIDOR	MEDIUM
6	18	Coffee	OFFICE	SHORT
6	19	Coffee	OFFICE	SHORT
6	20	Home Visit	HOME	MEDIUM
7	21	Lunch	CORRIDOR	MEDIUM
7	22	Coffee	OFFICE	SHORT
7	23	Coffee	OFFICE	SHORT
8	24	Lunch	CORRIDOR	MEDIUM
8	25	Coffee	OFFICE	SHORT
8	26	Coffee	OFFICE	SHORT
9	27	Travel together	OUTDOOR	LONG
10	28	Lunch	CORRIDOR	MEDIUM
10	29	Coffee	OFFICE	SHORT
10	30	Coffee	OFFICE	SHORT
11	31	Lunch	CORRIDOR	MEDIUM
11	32	Coffee	OFFICE	SHORT
11	33	Coffee	OFFICE	SHORT
12,13,14,15	ABSENCE	MONTH TRIP		
16	35	Lunch	CORRIDOR	MEDIUM
16	36	Coffee	OFFICE	SHORT
16	37	Coffee	OFFICE	SHORT

10.1.1. Trust Model Experimentation

In order to test the trust model, we have added the mentioned interactions in a database, the same that will receive the interactions from the communicating Sun SPOTS. Then, we calculated, using the formula mentioned at the section 9.2.4, the three contextual trusts (professional, public and private) at the end of each week and after every special event. It was considered as special events: the outdoor activity, the home visit, the trip that both agents have been together and the long absence periods.

Based on the results of the first weeks (shown in the Table 10) we noticed that the private trust value was being often degraded by the fact that we were calculating the average in consideration to all the events, instead of the ones where the private trust was being enhanced. The same effect was observed in the professional trust as being degraded after the outdoor activity.

Table 10: Results of tests for week 1-5 with old trust formula

End of week #	Trust Prof.	Trust Pub.	Trust Prvat.
1	1.633	1.167	0.467
2	1.633	1.167	0.467
3	1.633	1.167	0.467
After outdoor activity	1.47	1.19	0.49
4	1.508	1.185	0.485
5	2.091	1.601	0.651

Since the mentioned degradation effect does not seem to correspond with the real behavior of the trust, we adapted our trust calculation formula. The new formula, formula (2), is exactly like the previous one but it does not take into consideration the interactions where the contextual trust value is zero. As a result, in this second mode, the occurrence of events in a different contextual situation will not degrade the other context. For example, meeting a work friend at home would not degrade his professional trust despite the fact that the home contact does not add any professional trust enhancement.

$$T_{context} = \frac{0.7}{n'_{recent}} \times \left(\sum_{i=1}^{n'_{recent}} T'_{recent_i} \right) + \frac{0.25}{n'_{past}} \times \left(\sum_{i=1}^{n'_{past}} T'_{past_i} \right) + \frac{0.05}{n'_{old}} \times \left(\sum_{i=1}^{n'_{old}} T'_{old_i} \right) . \quad (2)$$

Where $T'_{time_i} = \{T_{time_i} | \, T_{time_i} \neq 0 \,\}$ and $n'_{time_i} = |T'_{time_i}|$

Using the formula (2) our results for the first five weeks were the following:

Table 11: Results of tests for week 1-5 with new trust formula

End of week #	Trust Prof.	Trust Pub.	Trust Prvat.
1	1.633	1.167	1.4
2	1.633	1.167	1.4
3	1.633	1.167	1.4
After outdoor activity	1.633	1.19	1.225
4	1.633	1.185	1.26
5	2.217	1.601	1.76

Continuing with the experiments, we noticed that the model proposed is too resistant to new punctual inputs. As the outdoor activity (represented in Table 11) and both the vacation and home visit (represented in Table 12) gives very little influence to the final trust values, despite they have a high trust associated with them. We tried to change a few of the values on the Trust Matrix (represented in Table 7) but there was very little impact on the results.

Table 12: Weeks 5-11 with trust formula (2)

End of week #	Trust Prof.	Trust Pub.	Trust Prvat.
5	2.217	1.601	1.76
6	2.183	1.537	1.76
After home visit	2.175	1.55	1.9
7	2.217	1.617	1.9
8	2.217	1.608	1.99
9 (trip week)	2.217	1.723	2.083
10	2.254	1.808	2.04
11	2.255	1.809	2.04

The mentioned resistance to new inputs is at some point good as it limits the trust variation from a biased or wrong input. In the other hand, it makes the trust value too much dependent on the user's routine. The ideal solution to address this would be to weight the different activities in comparison with the frequency associated with that specific activity, but this would require much more work characterizing the activities and defining the right weights. Therefore, we decided not to make any change on our model as the trust model is not our main focus.

Another thing we noticed from the analysis of Table 12 is that there are some other challenges in the categorization of the duration. The 3 day trip vacation could be modeled as a LARGE duration meeting as it is lasts more than 4 hours. Or we could, for example, consider as 3 meetings (once per day) lasting more than 4 hours. In the first case the resulting Private Trust would be of approximately 2.1 where in the second case it would be almost 2.4. This is something that should be considered in a real application.

Moreover, a real application should also define how to consider small interaction interruptions. For example, if two agents are working together in the same office, their interaction would suffer brief interruptions when each one goes to the toilet. The definition of the interval corresponding to the break of the connection will depend on the context of the interaction, the range of the sensing capabilities of the agents and the model itself (due to the effect mentioned in the previous paragraph). We will not address this question in the thesis, and we kept the experiments considering 3 interactions for the trip.

After introducing the vacations and the "year abroad" periods we noticed that the trust value is dramatically affected, as shown in Table 13. It is also possible to notice that the trust value is quickly recovered in the week after the vacation due to the fact that most of the final trust weight is based on the "recent" interactions. Moreover, if we adjust the threshold that corresponds for the "recent" interactions to more than a month, the trust values stay resistant to the vacation degradation (as shown in the last line of the Table 13).

Table 13: Trust after absence period

End of week #	Trust Prof.	Trust Pub.	Trust Prvat.
9 (3 days trip)	2.283	1.723	2.375
10	2.324	1.808	2.4
11	2.325	1.809	2.4
just before end of vacations	0.594	0.439	0.583
One week after vacation	1.994	2.073	1.75
Value considering a year abroad	0.118	0.090	0.111
Considering "recent" as less than 40 days	2.272	1.698	2.325

Those results lead us to reflect on how important it is to model the time degradation. Both the implementation of a time decreasing exponential factor as the time weight or the refinements of the weights and time frames for the time degradation categories could probably improve the results. Nevertheless, in order to discover the best weights, time frames characterization or the right exponential curve, more cases should be tested.

Another point that we observe on our experiments is that it would be beneficial to expand the contact from only presence-based to include also other communication formats such as letters, phone calls, e-mails, etc. This would be necessary in order to accurately represent cases such as when the agents separate from each other (due to a period abroad for example) but still keep contact and trust. It is important to capture the real life interactions, but the virtual ones should be considered as well.

The model with the second formula was able to represent somehow well the trust. In the beginning (Table 11), they share a professional trust in between "knowing each other from work" and "working in the same department", while privately they are close to "acquaintance" and publicly to "know each other". At that moment, the professional result is well represented. There are not enough events yet to evaluate well the other trusts, although maybe the public trust could have already been higher. After the three day trip together (Table 13), the private trust has been increased to a level between acquaintance and close friend, but the public is still far from the “share common interest” level. We considered changing the inferences values from the public trust, but the problem is that this could raise too much the public trust for people that just coincidently join public activities such as: people that takes the same buses or people at a company party offered to all employees. For the inference table, we conclude that there should be some changes there, but many other cases should be first evaluated.

At last we also noticed that the strategy of using just the average of the interactions must be adapted in a real implementation. When just computing the average, we end up with cases where the occurrence of additional events of small trust enhancement reduces the overall trust instead of increasing it. For example, with this model a single MEDIUM duration meeting would enhance less the trust then a MEDIUM duration meeting followed by two SHORT duration meetings. The interactions should result in a trust enhancement. This could be arranged by, for example, inserting the number of interactions either to the formula or as an additional context on the trust table.

Nevertheless, we finalize our trust logic tests in this point. We notice that there is a large space for improvements in the logic, but we will leave those for other researchers that may want to focus on the trust inference calculation from physical sensing.

10.2. Prototype Experimentation

We have conducted tests of the sensing capabilities during the whole development process: from the evaluation of the sensors with the Demo Projects from the Sun SPOTs to the individual testing of each thread. Thus, the last test remaining was to perform a small field test with the Sun SPOTs and being able to feed some data to the database, which could be later used to infer the trust.

For the experiments, we had two free-range sunspots, running the seamless trust relationship application mentioned in the last section, acting as the agents and one basestation that acted as the location provider and database proxy. The duration context was adapted, as described in Table 14.

Table 14: Experiment Adapted Duration Contextualization

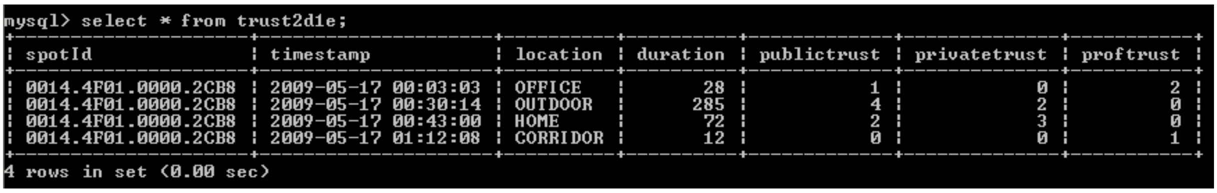
Duration Range	*Context*
Less than 15 seconds	PASSBY
From 15 seconds to 50 seconds	SHORT
From 50 seconds to 4 minutes	MEDIUM
More than 4 minutes	LARGE

The tests were performed by having a basestation and one of the free ranges SPOTs static in a room; and, entering the room with the second free-range SPOT for the specific duration. The basestation was reconfigured each time that the contextual location needed to be changed.

We have made tests in order to be able to capture the 4 different contextual locations and the 4 different duration types, as in Table 15. The test success is proved by the correctness of the trust values in the database as shown in Figure 27.

Table 15: Location and duration categories tests

Contextual Location	*Duration Category tested*
Home	Medium
Office	Short
Corridor	Drop by
Outdoor	Long

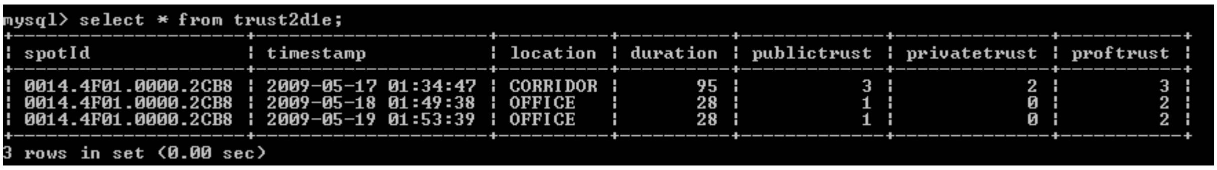


```
mysql> select * from trust2d1e;
+---------------------+---------------------+----------+----------+-------------+--------------+-----------+
| spotId              | timestamp           | location | duration | publictrust | privatetrust | proftrust |
+---------------------+---------------------+----------+----------+-------------+--------------+-----------+
| 0014.4F01.0000.2CB8 | 2009-05-17 00:03:03 | OFFICE   |       28 |           1 |            0 |         2 |
| 0014.4F01.0000.2CB8 | 2009-05-17 00:30:14 | OUTDOOR  |      285 |           4 |            2 |         0 |
| 0014.4F01.0000.2CB8 | 2009-05-17 00:43:00 | HOME     |       72 |           2 |            3 |         0 |
| 0014.4F01.0000.2CB8 | 2009-05-17 01:12:08 | CORRIDOR |       12 |           0 |            0 |         1 |
+---------------------+---------------------+----------+----------+-------------+--------------+-----------+
4 rows in set (0.00 sec)
```

Figure 27: Duration X Location X Trust in Database

Then, afterwards we made new tests capturing data emulating only the first week of the scenario described on the section 10.1. In other words, we made a test of MEDIUM duration in the CORRIDOR and two tests of SHORT duration on the OFFICE context. The timestamps of the tests were emulated as three different days by modifying the timestamp in the application and adding one day to it after each test.

The resulting data in the database can be seen in Figure 28. And, as expected, the trust computation for the end of that week was the same as the values presented in both Table 10 and Table 11.



```
mysql> select * from trust2d1e;
+----------------------+---------------------+----------+----------+-------------+--------------+-----------+
| spotId               | timestamp           | location | duration | publictrust | privatetrust | proftrust |
+----------------------+---------------------+----------+----------+-------------+--------------+-----------+
| 0014.4F01.0000.2CB8  | 2009-05-17 01:34:47 | CORRIDOR |       95 |           3 |            2 |         3 |
| 0014.4F01.0000.2CB8  | 2009-05-18 01:49:38 | OFFICE   |       28 |           1 |            0 |         2 |
| 0014.4F01.0000.2CB8  | 2009-05-19 01:53:39 | OFFICE   |       28 |           1 |            0 |         2 |
+----------------------+---------------------+----------+----------+-------------+--------------+-----------+
3 rows in set (0.00 sec)
```

Figure 28: Tests emulating a week of sensing

10.3. Evaluation

After performing the experiments, it is clear that there is a great and feasible potential for using what will be the Future SIM to build trust. We have successfully demonstrated how some of the sensors, which should be part of the Future SIM, can seamless capture data about people's relation and that this data can be used towards enhancing the accuracy of a trust network.

We managed to get some correct trust assumptions with a simple trust model and just one scenario. However, during the experiments we have been able to point flaws or imperfections on the model and we have proposed possible solutions to respond to them. We further recommend a conjunct work between sociologist and IT professionals to achieve a more realistic and accurate trust model, as we also recommend the choice of more cases and a real simulation to test the model.

We also recommend the usage of the Sun SPOTs as a platform to emulate a Future SIM. It does lack a real GUI and it lacks the connections with a mobile and a more widely deployed location sensor as Zigbee location mechanisms are still in early stage. However, it is possible to attach other pieces of hardware to it and the hardware specifications of the Sun SPOT are quite close to the ones of the SIM.

11. Conclusion

This study was performed in order to explore and demonstrate some of the capabilities of future SIM cards. It reviews the potential introduced by the potential of the Future SIM as a highly connected and secure context-aware platform and an identity device. We emphasize on the new usages towards trust and we prototype a new application that uses hard sensed evidences to support trust.

The recent hardware and software advances in the SIM card are shaping it into a complete and powerful networked embedded device. If we sum up the latest R&D publications on the development of processing power, memory (high-end cards), software platform (*JavaCard* 3.0 and SCWS) and communication capabilities (NFC, IPSIM, sensors, wireless interfaces) of the SIM, we achieve a Future SIM that could possibly supply all services (except the user interface) that the mobile devices serve today, but with the secure architecture inherited from the smart cards.

This Future SIM is highly connected, secure and it is not restricted to mobile phones, as they are already being connected with different devices in the M2M market. Moreover, it has embedded sensors, which together with the device in which the SIM is connected, grants a context-awareness capability to the SIM.

The SIM and other smart cards are already carry identity information in the shape of certificated, biometric data or business specific identifications, such as national ID numbers or MSISDN. It is part of complex Identity Management architectures where the identity application successfully share the same space as other applications (as in the case of the FINEID and MyKad); or where it conserves the user privacy through the seamless creation of pseudonyms like the Austrian ID; or through the roaming capabilities of the SIM.

The IdM architectures have also evolved in mean time by the introduction of open standards such as WS-* and SAML, and the creation of frameworks such as CardSpace, Higgins and the SIM IdM Framework proposed by the GSMA. Those frameworks separate the roles of identity providers and service providers, but they also specify methods for allowing federation, management of several identities, SSO and attribute sharing. The multi-application capabilities of the SIM and the security inherited by the GlobalPlatform Security Domains position the SIM as a perfect identity repository and selector. Moreover the new connectivity expands the range of interaction of those identities, and the context sensing allows a transparent identity selection mechanism. An extra argument to support the usage of the SIM as a central identity repository is endorsed by the support of the GSMA to use the same open standards used by the industry IdM frameworks.

By revising the identity role played by smart cards and specially the SIM, we noticed that it already acts as a trust component in a few policy-based trust schemes endorsed by SIM card digital signatures, certificates or by SLAs and other agreements established between the card and applications providers. Nevertheless, we see that it can greatly contribute to reputation based schemes as well. There, the future SIM can provide hard evidences to contextualize trust relations, to prove the capabilities of subjects towards a certain context, besides adding security to the storage and transitivity of the trust.

We identified a great range of cases that could go from avoiding falsification of items, to providing security and connectivity to objects in the internet of things, seamless context provisioning and identity and attribute management, besides adding hard evidences to context attributes and relations.

We prototyped an application implementing the case of a seamless trust builder where the Future SIM would be in charge of identifying the actors, sensing their interaction and defining trust values to their relations. As far as we know, it is one of the first attempts to represent contextual trust based on real interactions.

We have set a simple trust model that just took into account the timestamp, location and duration of the contact between the agents. However, besides those parameters, the future SIM could provide information based on identity profiles, or about the current activities of the user based on the use of his mobile or SIM. By having access to all this contextual information, much richer seamless models can be built.

Despite the fact that our trust model was only introduced as a prototypical implementation, we carried out a simulation of a simple case emulating the interaction between two co-workers. For our experiments we used the Sun SPOTs as the platform representing the future SIM. We managed to generate symbolic values for professional, private and public trust on a defined scenario. We noticed, as well, that it is important to somehow add to the model the sensing of the virtual activities as they have a relevant weight in the trust as well. Moreover, we detected some challenges in the calculation of this value such as the definition of thresholds and characterization of the activities and we recommend a conjunct work with professionals from the psychology and sociology fields to do so.

Despite our model's limitations, we believe that some more sensing inputs, changes in the logic and practical experiments can enhance it to a reliable seamless trust builder.

11.1. Future Work

This thesis is complete in the range of its scope. However, some of the results obtained are just in the conception or theoretical level and could be implemented in the real world. Or, the implementation done in the thesis could be revised and enhanced.

Practically, other researchers could use the assessed future SIM, IdM and trust capabilities to develop the identified applications towards countering falsification, connecting objects, acquiring hard contextual evidences or seamless managing identities and attributes.

They also could take the prototype we developed for the seamless trust builder and enhance its logic by adding unused sensed information such as identity profile attributes activities and motion. Or, they could perform a behavioral and relationship study in order to find the best values for our trust formula or to create a new formula. Moreover, a real simulation could be done in order to further identify the weak points of our model and to try to identify possible enhancements to the trust calculation.

Bibliography

3GPP Technical Specification Group Services and System Aspects. (2008). *Interworking of Liberty Alliance Identity Federation Framework (ID-FF), Identity Web Services Framework (ID-WSF) and Generic Authentication Architecture (GAA)(Release 8).* ETSI.

Abdul-Rahman, A., & Hailes, S. (2000). Supporting Trust in Virtual Communities. *HICSS '00: Proceedings of the 33rd Hawaii International Conference on System Sciences-Volume 6* (p. 6007). Washington, DC, USA: IEEE Computer Society.

Alani, H., Kalfoglou, Y., & Shadbolt, N. (2004). Trust strategies for the semantic web. *Proceedings of the Trust, Security and Reputation Workshop at the ISWC04* , 78-85.

AlNemr, R., & Meinel, C. (2008). Getting More from Reputation Systems: A Context-Aware Reputation Framework Based on Trust Centers and Agent Lists. *The Third International Multi-Conference on Computing in the Global Information Technology* , 137-142.

Anderson, R., & Kuhn, M. (1996). Tamper Resistance - A Cautionary Note. *PROCEEDINGS OF THE 2ND WORKSHOP ON ELECTRONIC COMMERCE*, (pp. 1-11). Oakland, California.

Ann Golbeck, J. (2005). Phd Thesis: Computing and applying trust in web-based social networks . University of Maryland, College Park.

Arora, S. (2008). *M.Sc. thesis: Review and Analysis of Current and Future European e-ID Schemes.* Roal Holloway, University of London.

Artz, D., & Gil, Y. (2007). A survey of trust in computer science and the Semantic Web. *Web Semantics: Science, Services and Agents on the World Wide Web* , 58-71.

Beattie, R. (2005, April 11). *Linking Out after Two Years of Linked In*. Retrieved May 03, 2009, from Russell Beattie's Weblog: http://www.russellbeattie.com/notebook/1008411.html

Beigl, M., Krohn, A., Zimmer, T., & Decker, C. (2004). Typical Sensors needed in Ubiquitous and Pervasive Computing. *Proceedings of the First International Workshop on Networked Sensing Systems*, (pp. 153-158).

Bernabeu, G. (2007). GlobalPlatform - the future of mobile payments. *Card Technology Today* , 9.

Bražinskas, R. (2008). *Msc. Thesis: Towards Context Awareness Using Mobile Sensors.* DTU.

Breslin, J., & Decker, S. (2007). The Future of Social Networks on the Internet: The Need for Semantics. *Internet Computing, IEEE* , 86-90.

Brickley, D., & Miller, L. (2007, November 2). *FOAF Vocabulary Specification 0.91*. Retrieved May 04, 2009, from FOAF: http://xmlns.com/foaf/spec/

Brown, K., & Mani, S. (2008). *Microsoft Code Name "Geneva" Framework Whitepaper for Developers* . Microsoft Corporation.

Cameron, K. (2005). *The Laws of Identity.* Microsoft.

Card Technology Today. (2008). Beefing up security with biometrics. *Card Technology Today* , 14-15.

Caverlee, J., Liu, L., & Webb, S. (2008). SocialTrust: Tamper-Resilient Trust Establishment in Online Communities. *JCDL '08: Proceedings of the 8th ACM/IEEE-CS joint conference on Digital libraries* (pp. 104-114). New York, NY, USA: ACM.

Cell Broadcast Forum. (2002). *Advantages and Services Using Cell Broadcast: Reaching Millions in Seconds.* Berne, Switzerland: Cell Broadcast Forum.

Chang, E., Hussain, F., & Dillon, T. (2006). *Trust and Reputation for Service-Oriented Environments: Technologies For Building Business Intelligence And Consumer Confidence.* Wiley.

Chappell, D. (2008). *Introducing "Geneva": an overview of the "Geneva" server, cardspace "Geneva", and the "Geneva" framework.* Chappell & Associates.

Dellarocas, C. (2003). The Digitization of Word of Mouth: Promise and Challenges of Online Feedback Mechanisms. *Management Science* , 1407-1424.

Dwyer, C., Hiltz, S. R., & Passerini, K. (2007). Trust and privacy concern within social networking sites: A comparison of Facebook and MySpace. *Proceedings of the Thirteenth Americas Conference on Information Systems* .

Eisl, F. (2004). *Msc. Thesis: Smart Card Security Services for an Open Application Environment used in Mobile Phones.* Lund: Lund University.

Elgan, M. (2008, November 26). *Why you can't trust 'friends' on Facebook*. Retrieved May 05, 2009, from ITWorld: http://www.itworld.com/security/58447/why-you-cant-trust-friends-facebook?page=0,0

Elsevier Ltd. (2009). ePassport status: 65+ countries now issuing. *Biometric Technology Today* , 3.

ETSI. (2009). *3GPP TR 33.812 V1.3.0: Feasibility Study on Remote Management of USIM Application on M2M Equipment; (Release 9).* ETSI.

ETSI. (2009). *ETSI TS 102 600 V7.4.0: Smart Cards; UICC-Terminal interface; Characteristics of the USB interface (Release 7).* Sophia Antipolis, France: ETSI.

Facebook. (2009). *Press Room: Facebook.* Retrieved May 2009, 05, from Facebook Web site: http://www.facebook.com/press/info.php?statistics

Finish Population Register Centre (VRK) . (2005). *FINEID - S2 VRK (PRC) CA-model and certificate contents v2.1.* Helsinki: Population Register Centre (VRK) .

Finish Population Register Centre (VRK). (2004). *Certification practice for a mobile citizen certificate used in mobile terminal equipment with subscription cards issued by TeliaSonera Finland Oyj v1.0.* Finish Population Register Centre (VRK).

Fogel, J., & Nehmad, E. (2009). Internet social network communities: Risk taking, trust, and privacy concerns. *Computers in Human Behavior* , 153-160.

France-Massey, T. (2005, September 14). MULTOS – the High Security Smart Card OS. London.

Gemalto. (2007). *Technical White Paper: Smart Card in IMS.* Gemalto.

Gert, T. (2007). *Phd Thesis: Electronic Voting over the Internet – an E-Government Speciality.* University of Technology Graz, Austria.

Geuer-Pollmann, C., & Claessens, J. (2005). Web services and web service security standards. *Elsevier Information Security Technical Report* , 15-24.

Giesecke & Devrient. (2007). *Smart Card Web Server: Merging the SIM and the World Wide Web.* Giesecke & Devrient.

Giesecke & Devrient. (2006). *White Paper: Bearer Independent Protocol (BIP).* Giesecke & Devrient.

Glass, B. (2000). *There in a Flash: Flash Memory for Embedded Systems.* Retrieved 03 12, 2009, from Embedded.com: http://www.embedded.com/98/9801spec.htm

Goldman, R. (2007). *A Sun SPOT Application Note: Using the LIS3L02AQ Accelerometer.*

GSM Association. (2008). *Identity Management Framework Document V1.1.* GSM Association.

GSM Association. (2007). *Mobile NFC technical guidelines - V2.* GSM Association.

Guthery, S. B., & Cronin, M. J. (2001). *Mobile Application Development with SMS and the SIM Toolkit.* McGraw-Hill Professional.

Handschuh, H., & Trichina, E. (2007). High Density Smart Cards: New Security Challenges and Applications. In N. Pohlmann, H. Reimer, & W. Schneider, *Securing Electronic Business Processes* (pp. 251-259). GWV-Vieweg.

Hendry, M. (2001). *Smart Card Security and Applications.* Norwood, MA, USA: Artech House, Inc.

Hristova, A. (2008). *Msc Thesis: Conceptualization and Design of a Context-aware Platform for User-centric Applications.* Madrid.

Jøsang, A., Hayward, R., & Pope, S. (2006). Trust network analysis with subjective logic. *ACSC '06: Proceedings of the 29th Australasian Computer Science Conference* (pp. 85-94). Hobart, Australia: Australian Computer Society, Inc.

Jurgensen, T. M., & Guthery, S. B. (2002). *Smart Cards: The Developer's Toolkit* . Prentice Hall PTR.

Karantonis, D. M., Narayanan, M. R., Mathie, M., Lovell, N. H., & Celler, B. G. (2006). Implementation of a Real-Time Human Movement Classifier Using a Triaxial Accelerometer for Ambulatory Monitoring. *IEEE Transactions on Information Technology in Biomedicine* , 156-167.

Karl, H., & Willig, A. (2005). *Protocols and Architectures for Wireless Sensor Networks.* Wiley.

Karri, R., Wu, K., Mishra, P., & Kim, Y. (2001). Fault-Based Side-Channel Cryptanalysis Tolerant Rijndael Symmetric Block Cipher Architecture. *DFT '01: Proceedings of the 16th IEEE International Symposium on Defect and Fault-Tolerance in VLSI Systems* (pp. 427-435). Washington, DC, USA: IEEE Computer Society.

Korpipää, P., Mäntyjärvi, J., Kela, J., Keränen, H., & Malm, E. (2003). Managing context information in mobile devices. *Pervasive Computing, IEEE* , 42-51.

Liersch, I. (2008). Id Cards and Passports. In K. Mayes, & K. Markantonakis, *Smart Cards, Tokens, Security and Applications* (pp. 323-345). Springer US.

Linn, J. (2000). *Trust Models and Management in Public-Key Infrastructures.* Bedford, MA USA: RSA Laboratories.

Looa, W., Yeowa, P. H., & Chongb, S. (2009). User acceptance of Malaysian government multipurpose smartcard applications. *Government Information Quarterly* , 358-367.

Lowton, M., Brown, J., & Finney, J. (2006). Finding NEMO: On the Accuracy of Inferring Location in IEEE 802.15.4 Networks. *REALWSN '06. Workshop on Real-World Wireless Sensor Networks* , 1-5.

Lu, H. K. (2007). Network smart card review and analysis. *Computer Networks: The International Journal of Computer and Telecommunications Networking* , 2234-2248.

Mardiks, E. (2005). *USB vs. MMC as High-Speed Interface from SIM Cards.* M-Systems Flash Disk Pioneers Ltd.

Markantonakis, K., & Mayes, K. (2003). An overview of the GlobalPlatform smart card specification. *Information Security Technical Report* , 17-29.

Markantonakis, K., Mayes, K., Tunstall, M., Sauveron, D., & Piper, F. (2007). Smart Card Security. In N. Nedjah, A. Abraham, & L. d. Mourelle, *Computational Intelligence in Information Assurance and Security* (pp. 201-233). Berlin: Springer Berlin / Heidelberg.

Massa, P. (2007). A Survey of Trust Use and Modeling in Real Online Systems. ITC-IRST, Italy.

Meyer, S., & Rakotonirainy, A. (2003). A survey of research on context-aware homes. *CRPITS '03: Proceedings of the Australasian information security workshop conference on ACSW frontiers 2003* (pp. 159-168). Darlinghurst, Australia: Australian Computer Society, Inc.

Milgram, S. (1967). The Small World Problem. *Psychology Today* , 61-67.

OASIS. (2005). *SAML V2.0 Executive Overview Committee Draft 01.* OASIS.

OASIS. (2006). *Security Assertion Markup Language (SAML) V2.0 Technical Overview; Working Draft 10.* OASIS.

OASIS Web Service Secure Exchange TC . (2007). *WS-Trust 1.3: OASIS Standard .* OASIS.

OASIS Web Service Secure Exchange TC. (2007, March 19). WS-Trust 1.3 Oasis Standard Specification.

O'Dell, M., O'Dell, R., Wattenhofer, M., & Wattenhofer, R. (2005). Lost in Space Or Positioning in Sensor Networks. *Workshop on Real-World Wireless Sensor Networks (REALWSN)*, (pp. 1-5). Stockholm, Sweden.

OMA. (2009). *Smart Card Web Service Standard Version 1.1.* OMA.

OMA. (2008). *Smartcard Web Server Enabler Architecture V1.0.* OMA.

Oostdijk, M., & Warnier, M. (2003). *On the combination of Java Card Remote Method Invocation.* Nijmegen Institute for Computer and Information Sciences.

Ortiz, C. E. (2008, June). *An Introduction to Near-Field Communication and the Contactless Communication API.* Retrieved March 13, 2009, from Sun Developer Network: http://java.sun.com/developer/technicalArticles/javame/nfc/

Payez Mobile. (2007). *Mobile contactless payment: Major industry players launch joint field trial.* Paris: Payez Mobile.

Pichler, A., & Hrachovec, H. (2008). *Wittgenstein and the Philosophy of Information.* Ontos Verlag.

Rankl, W., & Effing, W. (2004). *Smart card handbook.* John Wiley and Sons.

Renaudin, M., Bouesse, F., Proust, P., Tual, J. P., Sourgen, L., & .Germain, F. (2004). High Security Smartcards. *DATE '04: Proceedings of the conference on Design, automation and test in Europe* (p. 10228). Washington, DC, USA: IEEE Computer Society.

Riley, S. (2006, February 14). *It's Me, and Here's My Proof: Why Identity and Authentication Must Remain Distinct.* Retrieved March 22, 2009, from Microsoft TechNet: http://technet.microsoft.com/en-us/library/cc512578.aspx

Roessler, T. G., Posch, R., & Hayat, A. (2005). Giving an Interoperable Solution for Incorporating Foreign eIDs in Austrian E-Government. *Proceedings of IDABC-Conference 2005.*

Sanfey, A. G. (2007). Social Decision-Making: Insights from Game Theory and Neuroscience. *Science* , 598-602.

Schmidt, A., & Laerhoven, K. V. (2001). How to build smart appliances. *IEEE Personal Communications* , 66-71.

Schmidt, A., Aidoo, K. A., Takaluoma, A., Tuomela, U., Van Laerhoven, K., & Van de Velde, W. (1999). Advanced Interaction in Context. *Lecture Notes in Computer Science* .

Schmidt, S., Steele, R., Dillon, T. S., & Chang, E. (2007). Fuzzy trust evaluation and credibility development in multi-agent systems. *Applied Soft Computing* , 492-505.

Serrano, M. (2008). *Management and Context Integration based on Ontologies for Pervasive Service Operations in Autonomic Communications Systems.* Universitat Politècnica de Catalunya.

Six Apart. (2007, September 20). *We Are Opening the Social Graph*. Retrieved May 05, 2009, from Six Apart Website: http://www.sixapart.com/blog/2007/09/were_opening_th.html

Slagmolen, B., & Pastors, A. (2000). *Case Study Itafit: Finnish Electronic Identity Card – Fineid Card.* Itafit.

Smith, R. B., Cifuentes, C., & Simon, D. (2005). *Enabling Java for Small Wireless Devices with Squawk and SpotWorld.* Sun Microsystems.

Stutzman, F. (2006). An Evaluation of Identity-Sharing Behavior in Social Network Communities. *Journal of the International Digital Media and Arts Association* .

Sun Labs. (2008). *Sun™ SPOT Owner's Manual: Blue Release 4.0.* Sun Microsystems, Inc.

Sun Microsystems. (2009). *Sun SPOT World: Getting Started With Sun SPOTs*. Retrieved March 31, 2009, from Sun SPOT World: http://www.sunspotworld.com/GettingStarted/

Sun Microsystems. (2009). *Sun SPOT World: Product Specification*. Retrieved March 31, 2009, from Sun SPOT World: http://www.sunspotworld.com/products/

Sun Microsystems. (2008, August 21). *Sun™ Small Programmable Object Technology (Sun SPOT) Developer's Guide.* Retrieved March 31, 2009, from Sun SPOT World Web Site: https://www.sunspotworld.com/docs/Blue/spot-developers-guide.pdf

Sun Microsystems, Inc. (n.d.). Release Notes: Java Card Specifications Version 2.2.2 .

Sun Microsystems, Inc. (2008, March). Runtime Environment Specification: Java Card™ Platform, Version 3.0, Connected Edition.

Taherian, M., Amini, M., & Jalili, R. (2008). Trust Inference in Web-Based Social Networks Using Resistive Networks. *ICIW '08: Proceedings of the 2008 Third International Conference on Internet and Web Applications and Services* (pp. 233-238). Washington, DC, USA: IEEE Computer Society.

Techradar. (2008, January 11). *TechRadar Post: Facebook,Myspace Statistics*. Retrieved May 01, 2009, from TechRadar Blog: http://techradar1.wordpress.com/2008/01/11/facebookmyspace-statistics/

Toshiba America Electronic Components, Inc. (2006). *NAND vs. NOR Flash Memory: Technology Overview.* Irvine, CA: Toshiba.

Unisys. (2008). *Malaysia Smart Card Delivering Citizen Services Faster.* Unisys .

Veugelen, W., & Gilis, T. (2007). *Msc. Thesis: Identity management with Windows CardSpace and Electronic Identity cards.* Stockholm, Sweden: KTH.

Zennaro, M., Ntareme, H., & Bagula, A. (2008). Experimental evaluation of temporal and energy characteristics of an outdoor sensor network. *Mobility '08: Proceedings of the International Conference on Mobile Technology, Applications, and Systems* (pp. 1-5). Yilan, Taiwan: ACM.