# TRUST AND PRIVACY MANAGEMENT SUPPORT FOR CONTEXT-AWARE SERVICE PLATFORMS

RICARDO NEISSE

# Trust and Privacy Management Support for Context-Aware Service Platforms

# Trust and Privacy Management Support for Context-Aware Service Platforms

*Ricardo Neisse*



Enschede, The Netherlands, 2012

CTIT Ph.D.-Thesis Series, No. 11-216
SIKS Dissertation Series No. 2012-09

*Cover Illustration*: "The Privacy Eye" by Fernanda Neisse
*Cover Effects*: Ricardo Neisse
*Book Design*: Lidwien van de Wijngaert and Henri ter Hofte
*Printing*: Ipskamp, Enschede, the Netherlands

*Graduation committee:*

| | |
|---|---|
| *Chairman, Secretary:* | prof.dr.ir. A.J. Mouthaan (University of Twente) |
| *Promotor:* | prof.dr. R. J. Wieringa (University of Twente) |
| *Assistant Promotors:* | dr.ir. M. J. van Sinderen (University of Twente) |
| | dr. M. Wegdam (University of Twente) |
| *Members:* | dr. B. Crispo (University of Trento) |
| | prof.dr. S. Etalle (University of Twente/TU Eindhoven) |
| | prof.dr. W. Jonker (University of Twente) |
| | prof.dr. S. Katzenbeisser (Technical University of Darmstadt) |
| | dr. J.M. Seigneur (University of Geneva) |

CTIT Ph.D. Thesis Series No. 11-216
ISSN 1381-3617
Centre for Telematics and Information Technology
P.O. Box 217, 7500 AE
Enschede, The Netherlands

SIKS Dissertation Series No. 2012-09
The research reported in this thesis has been carried out under the auspices of SIKS, the Dutch Research School for Information and Knowledge Systems.

ISBN 978-90-365-3336-2
http://dx.doi.org/10.3990/1.9789036533362

# Trust and Privacy Management Support for Context-Aware Service Platforms

PROEFSCHRIFT

ter verkrijging van
de graad van doctor aan de Universiteit Twente,
op gezag van de rector magnificus,
prof.dr. H. Brinksma,
volgens besluit van het College voor Promoties
in het openbaar te verdedigen
op vrijdag 30 maart 2012 om 12.45 uur

door
Ricardo Neisse
geboren op 09 maart 1979
te Carazinho, Rio Grande do Sul, Brazilië

Dit proefschrift is goedgekeurd door:
prof.dr. R. J. Wieringa (promotor) , dr.ir. M. J. van Sinderen (assistent-promotor),
en dr. M. Wegdam (assistent-promotor)

# Abstract

In a context-aware service platform, service providers adapt their services to the current situation of the service users using context information retrieved from context information providers. In such a service provisioning platform, important trust and privacy issues arise, because different entities responsible for different tasks have to collaborate in the provisioning of the services. Context information is privacy sensitive by nature, making the communication and processing of this information a potential privacy threat.

The main goal of this thesis is to learn how to support users and providers of context-aware services in managing the trade-off between privacy protection and context-based service adaptation. More and more precise context information retrieved from trustworthy context information providers allows context-aware service provider to adapt their services more reliably. However, more and more precise context information also means a higher risk for the service users in case of a privacy violation.

User acceptance of context-aware services depends on the users' perception of how the context-aware service platforms deal with their privacy. Users of context-aware services need to control who is authorized to access their context information, and how their context information is communicated and processed after the access is granted. Providing users with control over their privacy is especially difficult in context-aware service platforms, since users' privacy desires are personalized to the context situation these users are in. Users have, for example, different wishes regarding privacy of their health data when they are being treated in a hospital than when they are in their working environment.

For users to feel in control of their privacy, the mere specification of their privacy preferences is not enough in the trade-off between privacy and context-aware service adaptation. Users must also be confident that the specified privacy preferences are being enforced by the entities of the service platform that are responsible for communication and processing their context information, such as the context information providers. Trust, therefore, is an integral part of the users' privacy concerns in context-aware service platforms.

In the trade-off we address in this thesis, context-aware service providers are more concerned with their capability of providing reliable context-based service adaptation because this is their primary business goal. To be able to reliably adapt, service providers depend on the trustworthiness of the context information providers that provide the context information about the service users. Privacy issues are also important for service providers because the reporting in the media of privacy incidents involving their service provisioning infrastructure also impact their primary business due to the loss of reputation.

Existing trust and privacy solutions targeted at context-aware service platforms fail to address the different trust aspects and dependencies between the entities participating in a context-aware service provisioning platform. Existing solutions focus on at most one trust aspect at a time, for example, privacy enforcement or identity certification, and do not consider dependencies between the different aspects that are present in the trade-off we address in this thesis. Other concerns of users and service providers such as reliability of the context-aware service adaptation, or the relationship between quality aspects of the context information and trust are not addressed by existing solutions in an integrated way. Furthermore, existing trust and privacy management solutions for context-aware service platforms offer poor support for personalized context-based privacy management.

In this thesis we present the analysis, design, implementation, and evaluation of a trust and privacy preferences management solution to support service users and service providers of a context-aware service platform. The functionality of this solution consists of three major contributions that focus on trust and privacy issues from the perspective of users and service providers.

The first major contribution of this thesis is a trust-based selection mechanism that support users of context-aware services in selecting trustworthy service providers to interact with. This mechanism supports the users in this selection process, taking into account the users' goals, trust beliefs, and the trust dependencies between the service users and the entities that collaborate in the context-aware service provisioning. The service users' goals we use as input to our mechanism are related to the trade-off between privacy protection and context-aware service adaptation.

The second major contribution of this thesis is a trust-based selection mechanism that support context-aware service providers in selecting trustworthy context information providers. This mechanism supports service providers in selecting context information providers taking into account their trustworthiness to provide context information about a specific user and quality level. This mechanisms contributes to the im-

provement of the context-based adaptation capabilities of context-aware service providers.

The third major contribution of this thesis is a generic context-based policy management concept called a Context-Aware Management Domain (CAMD). The CAMD concept is used by us in our case studies to support the specification of trust and privacy policies by means of context-based authorizations and obligations. Our CAMD concept is realized using policy-based management, and uses context information as input for the policy management task. The objective of our CAMD concept is to support users and system administrators in managing policies aimed at controlling who is authorized to access the users' context information at what quality level, and which actions these entities are obliged to execute after access is granted. One example of a privacy obligation is to delete all location information about a user after a context condition is satisfied.

We have evaluated the technical feasibility of our contributions through case studies and prototype implementations. We have also evaluated the usability and usefulness aspects of our contributions from a user perspective through an user survey. Our technical and user survey evaluations show that our solutions are technically feasible and that the majority of the survey participants were able to understand and believe that our contributions are useful. Furthermore, our technical feasibility and user survey evaluations contribute to increased knowledge about the trust and privacy requirements of a context-aware service platform with examples of context-based policies and user goals when using a context-aware service.

# Acknowledgments

I would like to start by thanking my promoter Roel Wieringa and my daily supervisors Maarten Wegdam and Marten van Sinderen for their support and guidance. I have learned a lot from you from many different perspectives. Roel got to know my research nearly in the end, when I moved from the ASNA to the IS group, and I was impressed by the feedback I received from him on the content and methodological levels. Marten was a constant source of motivation and inspiration, all discussions with him during my PhD work helped me to look at my research from different perspectives and think outside of the box. Maarten showed me how to structure and take a pragmatic view of my research with respect to real problems outside of the academic world.

I would like to thank the members of my defense committee dr. Bruno Crispo, prof. dr. Sandro Etalle, prof. dr. Willem Jonker, prof. dr. Stephan Katzenbeisser, and dr. Jean-Marc Seigneur. Thank you for spending your time reading my thesis and participating in my defense.

During my PhD work I have collaborated more closely with two outstanding researchers, Patrícia Dockhorn Costa and Gabriele Lenzini. I would like to thank you for the nice discussions and for you help shaping my PhD work.

Even before my PhD work, during my bachelors and masters, I was fortunate to work with Lisandro Granville. Since then, Lisandro has been a good friend and a mentor in my personal and professional life. Lisandro, thank you for the support and encouragement on pursuing an academic career.

From the University of Twente I would like to thank Belinda Jaarsma-Knol, Annelies Veldman-Klos and Suse Engbers. Belinda and Annelies were extremely helpful with all the arrangements and provided me with a very warm welcome when I first arrived in the Netherlands. I would like to thank my colleagues from the IS group, from the ASNA group, and from the AWARENESS research project. It was always a lot of fun to collaborate with you and also to meet you for social activities, parties, dinners, uitjes, lunches, and coffee breaks.

When I joined Fraunhofer in Kaiserslautern I was lucky to meet outstanding researchers and great persons. From all the people I have met in Fraunhofer I specially would like to thank Alexander Pretschner, Enrico Lovat, Prachi Kumari, and Sonnhild Namingha. Thank you for the nice time together, I have learned a lot from you.

I would like to thank the Brazilian National Council of Research and Development (CNPq) that provided me support during my whole academic career including my PhD work.

During my PhD studies I have met a lot of amazing people who definitely deserve a special mention. I would like to thank my friends Patrícia and João Paulo, Luiz Olavo and Luciana, Giancarlo and Renata, Giovane, Laura, Rodrigo, Rafael and Sasha, Tiago and Liga, Eduardo, Kamran, Tom, Hailiang, Raphael and Dulcinéia, Enrico and Simona, Diana, Aleks, Sharon, and Márcia. Guys, you are all very special friends to me. Special thanks of course to my paranymphs and great friends Luiz Olavo and Giovane for the help in the organization of my PhD defense.

The following two paragraphs are the acknowledgments to my family in Brazilian Portuguese and Polish languages.

Minha mãe Zelita, minha irmã Fernanda, e meu cunhado Geancarlo por estarem sempre presentes mesmo com a distância e com um oceano entre a gente. Agradecimentos especiais para minha irmã pelo fabuloso trabalho de arte na capa deste livro. Mãe, obrigado pelo incentivo e motivação nos meus estudos, grande parte das minhas conquistas não teriam sido possíveis sem você. Sei que posso contar contigo sempre!

Chcę bardzo podziękować mojej polskiej rodzinie, tacie Kasi Włodkowi, mamie Bożenie, siostrom Jagodzie i Marysi za serdeczne przyjęcie mnie do rodziny i za wspaniałe, wspólnie spędzone chwile.

The most important thank you goes to my wife Kasia, she has always been there for me and gave me all the love and support to finish my PhD thesis. I'm lucky to have met you and I'm looking forward to spend the rest of my life with you, enjoying the best life has to offer. This book is dedicated to you babe, love you!

Ricardo Neisse, Kaiserslautern, March 9th 2012.

# Contents

Chapter 1

# Introduction

This thesis proposes trust and privacy management extensions for context-aware service platforms in order to increase users' control over their privacy and to improve the context-based adaptation capability of context-aware service providers.

This chapter is organized as follows. Section 1.1 presents background information on context-aware service platforms. Section 1.2 motivates the research and describes the research problems we address. Section 1.3 presents the main goal of this thesis and its sub-goals. Section 1.4 explains the approach we follow. Section 1.5 describes the scope of this thesis. Section 1.6 ends this chapter with a concise description of the thesis structure.

## 1.1 Background

The evolution of computers, sensors, and wireless networks is leading to the pervasive availability of computers and information technology (IT) services in people's daily lives. With this evolution, it became technically possible and economically viable to automatically measure detailed physical properties of objects and environments. Examples of these measurements are ambient temperature and geolocation coordinates that can be determined unobtrusively and without human intervention by sensors in a mobile phone.

The availability of these measured properties enables the realization of certain types of services, that adapt to specific user needs considering changes on these objects and environment properties. These new types of services are known as context-aware services [37]. The measured properties that are associated to a service end-user describe the service *user context*.

The benefit of context-aware services for users depends on how well the service adaptation fulfill the user needs, which in its turn depends on how well the user context is measured. The added value of context-aware services, compared to traditional IT services, lies in the automated context-based adaptation. Traditional IT services could also fulfill the service users' needs in specific situations by requesting manual user input. However, the effort required from users with manual input is too much to justify the adaptation benefits.

Examples of context-aware services are:

- A weather forecast service that automatically provides the weather forecast based on users' current location and his or her next planned or predicted destinations;
- An office service that helps colleagues working in a company to find each other faster when needed based on their location, scheduled appointments, and activities;
- A tourist guide service that personalizes the tourist advice for users based on their location, weather conditions, and traveling interests;
- A health service that chooses the most suitable and nearby caregivers to help a patient if needed based on the patient's real-time health data, the patient's location, and the caregivers' location and availability.

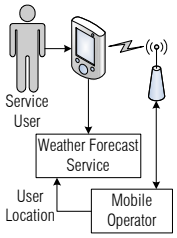



Figure *1-1* Context-aware weather service example

Service providers capable of providing context-aware services are not necessarily also responsible for capturing the context information of the service users. One example that illustrates this is a weather forecast service that automatically determines the city/region of the service user from the location of the GSM cell the user is currently connected to. In this example, presented in Figure *1-1*, the context information provider (in short, context provider) could be the user's mobile phone operator which is part of a different administrative domain than the weather forecast service provider.

For the weather forecast service, only the context information of the service user is relevant. However, other context-aware services may require context information about other entities than the user of the service. In the office service example, if a service user wants to find one specific employee, only the context information of that employee is important. Even though their privacy might be affected, employees might agree to use this service, and have their context information accessed, if they see a potential benefit in facilitating contact with other colleagues[1].

The weather forecast service and the office service examples illustrate the importance of the users' identity[2] for providing context-aware ser-

[1] In this thesis, we use the term context-aware service user to refer to the user who requests a context-aware service or whose context is relevant to the requested service.

[2] In this thesis we refer to the term *identity* meaning a *digital identity*.

vices. In the weather forecast service, not all identity attributes of the service user are important for the service provisioning as they are in the office service. In the office service, the true name of the entities is of crucial importance, since the service users are interested in context information about specific colleagues. For privacy reasons, the anonymization of the users' identities for services like the weather forecast service might be desired but not for all services.

Solutions for identity management such as identity federation and Single Sign On SSO are being used in context-aware service platforms to identify and manage the identities of context-aware service users [59]. These identity management solutions allow users to choose one of their identities and configure which services are authorized to access their profile information. Some of the existing solutions also provide support for full or partial anonymization of the users' identity and context information [10, 108]. These solutions contribute to reduce the privacy risks for service users.

The examples in this chapter illustrate that the provisioning of context-aware services depends on the cooperation of service providers, context providers, and identity providers. Each of these roles is responsible for specific tasks and might be located in different administrative domains. In order to facilitate this cooperation and reduce the complexity of the design, implementation, and deployment of context-aware services, context-aware service middleware has been developed [119].

Context-aware service middleware provides generic support for context acquisition, reasoning, and distribution to applications that provide context-aware services. In this thesis, we are particularly interested in developing extensions to existing context-aware service middleware to include trust and privacy management from the point of view of context-aware service users and context-aware service providers.

Figure *1-2*[3] shows the classification we defined for the existing trust and privacy management approaches and how they relate to each other. Our classification is inspired by the analysis of trust management approaches done by Gollmann [46] and on our own literature study. Existing work on trust management focuses on distributed authentication and authorization schemes, reputation and reliability measurements, and integrity and attestation using tamper proof hardware. Existing work on privacy management focuses on protection of personal data from unauthorized usage, reputation and reliability approaches that focus specifically on privacy protection issues, and approaches focusing on data obfuscation or quality degradation to reduce by design the observation capabilities of other entities and consequently the privacy risks.

[3] This classification also reflects the structure of Subsection 2.2 and 2.3 of this thesis.

The arrows in the Figure *1-2* shows the functional interrelation between the different trust and privacy management classes of work. Existing trust management approaches to support distributed authentication and authorization are related to privacy management approaches that support the protection of personal information from unauthorized usage because both classes focus on authorizations. Existing work focusing on reputation and reliability can not be clearly separated in privacy and trust management classes because most of the time the word trust is used interchangeably with privacy meaning trust with respect to privacy protection. Therefore, we show the interrelation between these two classes using a thicker arrow. Existing trust management approaches using tamper proof hardware for integrity checking and attestation are related to privacy management approaches because the provisioning of these integrity measurements contributes to increase the reputation and reliability with respect to privacy protection.

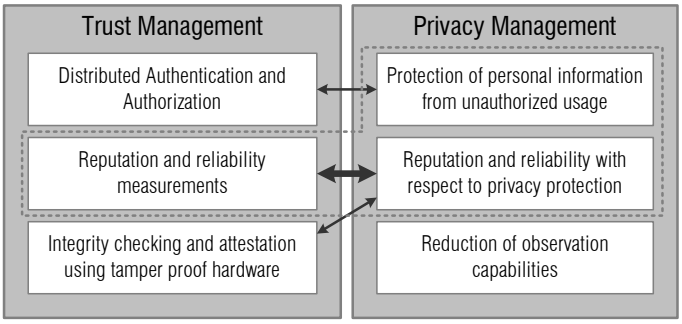



Figure *1-2* Classification of trust management and privacy management approaches

The focus of the work described in this thesis is on reputation and reliability aspects both for trust and privacy management, and on privacy management to protect personal data from unauthorized usage. This focus is highlighted with a dashed line polygon in Figure *1-2*. The other classes of privacy and trust management approaches are out of the scope of this thesis.

## 1.2 Motivation and Problem Description

In this section, we present the trust and privacy management problems of context-aware service platforms that are addressed in this thesis:

***Users Need to Feel in Control of Their Privacy to Accept Context-Aware Services***

Context is privacy-sensitive

Context-aware service platforms process and communicate context information related to the service users in order to increase the usability

and usefulness of the services provided. This particular feature of context-aware service platforms implies both an opportunity and a risk for the service users. The opportunity is the possibility of services customized to the service users' environment and needs. The risk for service users' is related to the possible privacy violations that may occur when privacy-sensitive user context information is processed and communicated by the entities that collaborate in the provisioning of context-aware services. The actual risk is that context information is used for purposes not agreed upon or intended by the service user, by entities collaborating in the service provisioning or by *outsiders* that are able to acquire the context information. Outsiders are not within the scope of this thesis.

User acceptance depends on the service users' feeling regarding privacy control

In this thesis, we address the privacy control support required for users when their context information is processed and communicated by context information providers to context-aware service providers. For privacy reasons, users will be more likely to use context-aware services if they feel in control of who accesses their context information and how their context information is used after it is released [67]. An example of user control is the possibility to select a privacy preference that states that only entities authenticated by the service user's company are authorized to access the context information. Furthermore, privacy preferences can also include obligations stating, for example, that all context information accessed by authorized entities should be deleted one day after it has been retrieved from the context information providers.

Privacy risks and privacy preferences requirements

The privacy risk is a problem for users of context-aware services mainly due to the privacy-sensitive nature of the users' context information, and the implicit gathering and combining of this information in a pervasive context-aware service provisioning environment. When context information is accessed by malicious entities, serious privacy violations and security infringements such as unauthorized user tracking, unauthorized sophisticated user profiling, and subsequent identity theft are enabled. To avoid these privacy violations and consequently user distrust in and abandonment of context-aware services, users should be able to specify their privacy preferences in a personalized and understandable manner.

Privacy preferences are context-based

Privacy preferences for context-aware service users differ from traditional privacy preferences because they depend on the context situation the users are in. For example, users might have different wishes regarding privacy when they are in a health emergency situation than when they are in a normal working day. For example, when an user is in a health emergency situation, he would be more willing to authorize indiscriminate access to all his context information. Furthermore, changes in context situations can trigger privacy obligations. For example, when the health emergency situation has passed, all the context information collected during the health emergency should be deleted.

Limitations of existing context-based privacy solutions

Solutions for privacy preferences management targeting context-aware service platforms [75, 29, 27, 28] offer poor support for personalized context-aware privacy management and do not address context-based privacy obligations. These approaches are also limited with respect to the privacy support they provide because they mostly focus only on access control aspects of privacy. Privacy in these approaches is defined by means of authorization rules using policies that govern the access to the user's context information. These context-based privacy management solutions have not been evaluated with respect to their usability, usefulness, and user acceptance. In Figure *1-2* these solutions are part of the class *Protection of personal information from unauthorized usage* in *Privacy Management*.

***Selection of Trustworthy Entities with Respect to Privacy Protection***

Trust and privacy requirements

The mere specification of their privacy preferences is not enough for users to accept using context-aware services. Users will only feel in control of their context information if they also trust that the privacy preferences they have specified are honored by the platform that handles and communicates their context information. Trust, therefore, plays an important role and is an integral part of the privacy concerns of context-aware service users.

Trust aspects for different roles

In a pervasive context-aware service platform, users may have a choice of different entities that are capable of collaborating and providing a specific context-aware service. The trustworthiness of a context-aware service depends on the trustworthiness of the entities that collaborate during the service provisioning process for different trust aspects, namely, context providers, service providers, and identity providers. For privacy reasons, users are likely to accept context-aware services only if they can trust that:

1. The context providers are releasing privacy-sensitive context information only to entities authorized by the user;
2. The service providers receiving context information are enforcing the users' privacy preferences.

Trust leads to user acceptance

The trustworthiness evaluation of a context-aware service depends on all the trust aspects mentioned above. Based on their trustworthiness evaluation users may decide to have more or less strict privacy preferences or may decide not to use a context-aware service at all. Existing solutions that address trust management for context-aware service platforms do not consider the dependencies between the different trust aspects for the entities that collaborate in the provisioning of the context-aware service. Trust management solutions for context-aware services [34, 28] adopt a simplistic approach where trust is claimed to be related to privacy but actually is related only to authorization policies or identity certification issues. In Figure *1-2* these solutions are part of the class *Reputation and*

*reliability measurements* in *Trust Management* and *Reputation and reliability measurements with respect to privacy protection* in *Privacy Management*.

Trust management mechanisms support users when they interact with unknown entities

Trust management support is important when users interact with entities that they do not know or with which they have not interacted before. In this case, a trust management mechanism for context-aware service platforms should support users in evaluating the entities' trustworthiness based on other trust sources such as trust recommendations. Furthermore, trust management solutions should support quantification of uncertainty about trustworthiness (e.g. [71]) when entities are unknown and recommendations about trustworthiness are not available. The major benefit of a trust management mechanism is a systematic trust support that enables users to assert the trustworthiness values of the known and unknown entities that collaborate in the provisioning of a context-aware service and to select the more adequate entities considering the entities trustworthiness, the users' goals and privacy risks.

***Selection of Trustworthy Entities with respect to Reliable Context-Based Service Adaptation***

Context-aware service providers depend on context providers to adapt reliably

From the context-aware service providers' point of view, the primary concern is not the privacy of the service users. Service providers are mainly concerned with their ability to provide a reliable context-based service adaptation meaning that the context-aware service adapts consistently to the users' current situation and needs. For some service providers privacy issues are also important because the reporting of privacy incidents in the media also impact their primary business due to the loss of reputation.

A service provider can only adapt consistently if the context information provider is trustworthy to reliably measure and communicate the context information values with the required quality characteristics. For this reason, service providers depend on the trustworthiness of the context information providers that provide the context information about the service users. The trustworthiness of the context providers is important also for users who value the context-aware adaptation even though they are less concern with their privacy being protected.

Users want a reliable context-aware service

For users that are concerned with a reliable context-based service adaptation, the trustworthiness of the service providers and context providers is important for trust aspects other than privacy. These users must trust that:

1. The context-aware service providers are providing reliable context information about them;
2. The service provider, by receiving reliable context information, is adapting reliably to their context and thereby providing a context-aware service with an added end-user value;
3. The identity providers are capable of correctly and reliably identifying them.

Trustworthiness of context information providers depends on trustworthiness of identity providers

Identities provided by trustworthy identity providers are required to ensure that the retrieved context information corresponds to the correct users' identity. A trustworthy identity provider is capable of reliably authenticate users and provide identity attributes that truly belong to the user holding the identity. A context-aware service user that uses an identity provided by an insecure identity provider might be more vulnerable to fake or incorrect context information being delivered due to the influence of malicious entities. An insecure identity provider might allow a malicious entity to create a fake identify and impersonate an user. Depending on the user and service provider goals - for instance, if both are interested in reliable service adaptation - the most trustworthy identity and context providers are preferred in order to increase the reliability of the context-based service adaptation.

Relationship between Quality of Context (QoC) and trust

The capability of service providers to reliably adapt their context-aware services depends on the trustworthiness of the context information providers. The trustworthiness of a context provider relates to the quality level of the context information provided by this provider, which can be quantified using Quality of Context (QoC) attributes [18]. Therefore, service providers should select context information providers considering their trustworthiness and the QoC level they support.

Trustworthy context information providers capable of providing context information at high QoC levels contribute to the reliability of context-based service adaptation, but it also increases the privacy risks of users. Privacy risks of users are increased because more precise and reliable information about them will be exposed in case of privacy violations. Despite being considered part of a QoC specification, trustworthiness is not concretely addressed in existing QoC modeling approaches [18, 113].

## 1.3 Research Goal and Sub-Goals

The main goal of this thesis is: *to learn how to support users and providers of context-aware services in managing the trade-off between privacy protection and context-based service adaptation*. The following sub-goals detail the main goal. We have classified each sub-goal as a knowledge or a design sub-goal [4]. The classification into design and knowledge is indicated between parentheses for each of the sub-goals:

– To research the role of trust in context-aware service platforms and to identify the relevant roles and trust relationships that influence the reliability of the context-based service adaptation and the protection of the users' privacy (*knowledge*);

[4] A knowledge sub-goal focuses on learning something new and a design sub-goal focuses on improving the way something is done. Most of the time, knowledge and design sub-goals are composed of sub-sub-goals of both types [122].

- To identify the relationship between trust and Quality of Context (QoC) (*knowledge*);
- To propose solutions to support service users in selecting trustworthy identity providers, service providers, and context providers. The objective of these solutions is to fulfill the users' goals in the trade-off between privacy protection and reliability of the context-based service adaptation (*design*);
- To propose solutions to support context-aware service providers in selecting trustworthy context providers. The objective of these solutions is to improve the reliability of the context-based service adaptation (*design*) and fulfill the service providers' goal of reliable context-based service adaptation;
- To propose solutions that allow users to manage their trust and privacy preferences, including support for *context-based* personalized authorizations and obligations (*design*);
- To validate the technical feasibility of our work through case studies and prototype implementations (*knowledge*);
- To evaluate if users need the solutions we propose, are able to understand the concepts involved (usability), and benefit from (usefulness) these concepts (*knowledge*). It is also part of this sub-goal to learn about trust and privacy preferences requirements of service users.

## 1.4 Approach

We followed the steps below in our research:

1. Research the state of the art on trust and privacy management in general for distributed systems and specific for context-aware service platforms;
2. Analyze the roles and responsibilities of each role in a context-aware service platform. The roles we consider are: service user, context owner, identity provider, service provider, and context provider;
3. Identify the trust relationships between the roles related to the trade-off between privacy and context-based service adaptation;
4. Design a trust management model, architecture, and selection mechanisms for context-aware service platforms considering our analysis of the trust relationships related to the trade-off between privacy and con-text-based service adaptation.
5. Design a *context-based* trust and privacy management solution that uses context as input for the trust and privacy management tasks and supports the specification and of *context-based* authorizations and obligations;

6. Conduct case studies and implement proof-of-concept prototypes to validate the technical feasibility of our contributions;
7. Validate the designed and implemented models and mechanisms from a user perspective through an interactive user survey.

## 1.5 Scope

The contributions proposed in this thesis focus on the trust aspects of context-aware service platforms from the points of view of context-aware service users and context-aware service providers. Our contributions are not aimed at supporting identity providers and context providers in managing their trust relationships with other entities.

Our trust and privacy management mechanism provides generic context-based management support for users using a policy-based approach. However, in our case studies and prototype implementations, we focus on personalized context-based usage control and trust management policies. The usage control policies comprise authorization and obligation policies for the users' context information specifying who has access to the users' context information and the rights and duties that should be enforced after the information is accessed. The trust management policies specify the possible evolution of the trustworthiness values according to specific conditions.

From the context-aware service user's point of view, we provide trust and privacy management support for users that are concerned with the goal of privacy protection or with the goal of reliability of context-based service adaptation. These goals are related to the trade-off between privacy protection and context-based service adaptation. The trust management mechanism we describe to support the selection of trustworthy context providers targets the reliability goal. It is not our objective in this thesis to support other user goals, for example, goals related low service cost or service availability.

From the service provider's point of view, we provide trust management support to allow the selection trustworthy context providers. We do not focus on trust aspects related to internal components of the context-aware service that might influence its capabilities of correctly adapting to he context of the users, such as optimized and reliable implementations of the service. We also do not support any external factor other than trustworthiness values associated with the identity providers and context providers.

Although extensible, the trust model we have developed is designed to support context-aware service users and service providers focusing on trust aspects related to identity provisioning, privacy enforcement, context information provisioning, and context-aware service provisioning.

## 1.6 Thesis Structure

The structure of this thesis follows the same structure as the approach described in the previous sub-section and is explained in the list below.

- *Chapter 2 - State of the Art.* This chapter presents a description of the state of the art in context-aware systems, trust management, and privacy management;
- *Chapter 3 - Trust-based Selection of Context and Service Providers.* This chapter presents the trust management model and mechanisms we propose for context-aware service platforms. In this chapter, we validate the technical feasibility of our contributions by means of a prototype implementation for supporting context-aware service users and providers. Our objective in this chapter is to support:
  1. Users of context-aware services in selecting trustworthy context-aware service providers considering the trade-off between privacy and context-based service adaptation;
  2. Service providers in selecting trustworthy context information providers considering the relation between trustworthiness and Quality of Context (QoC) that is also detailed in this chapter;
- *Chapter 4 - Context-Based Trust and Privacy Management.* This chapter presents a new approach for context-based management of personalized trust and privacy policies from the service user perspective. The main contribution of this chapter is a generic context-based policy management concept called Context-Aware Management Domain (CAMD) and its specific use in the management of context-based authorizations, privacy obligations, and trust management obligation policies. In this chapter, we validate the technical feasibility of our CAMD concept by means of two prototype implementations;
- *Chapter 5 - User Survey.* This chapter presents the user survey we conducted to validate the usability and usefulness of our trust and privacy management solutions from the service user's point of view. We describe the validation scenario, the method, our results, and the analysis of our results;
- *Chapter 6 - Conclusions.* This chapter summarizes the contributions of this thesis together with a critical analysis of these contributions. The critical analysis of the contributions leads to the identification of open issues that require further investigation.

Figure *1-3* graphically presents the structure of this thesis and how the chapters are related to each other. The solid lines represent the temporal precedence and the dashed lines show how the contents of the chapters are related.

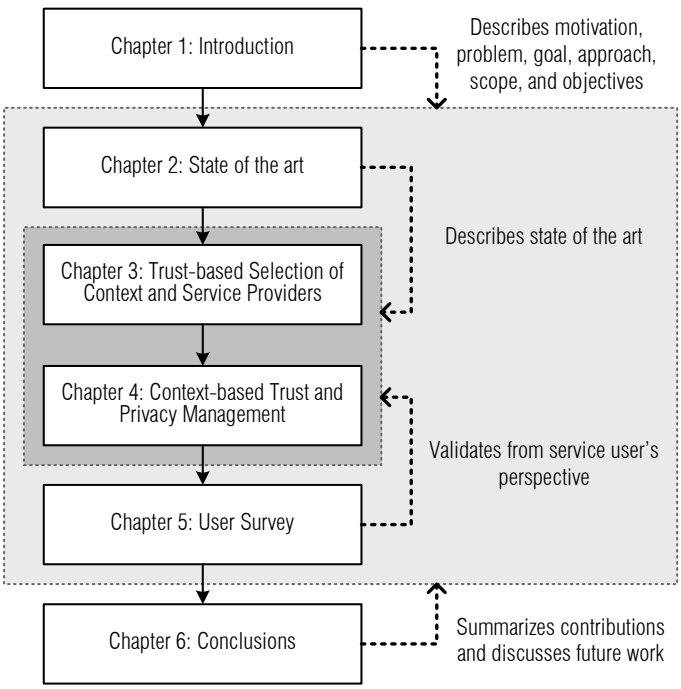



Figure *1-3* Thesis chapters

Chapter 2

# State of the Art

This chapter presents the state-of-the-art in context-aware systems, trust management, and privacy management from a social science and information technology point of view. The remainder of this chapter is organized as follows: Section 2.1 describes existing work in the area of context-aware systems; Section 2.2 presents trust management solutions; Section 2.3 discusses related work on privacy management focusing in the specification of privacy preferences and techniques to enable privacy by design; Section 2.4 ends this chapter with a summary and discussion.

## 2.1 Context-Aware Systems

Context awareness is the *ability of applications to utilize information about the user's environment (context) in order to tailor services to the user's current situation and needs* [39]. In this section we describe existing work on:

- Context information modeling to support context-aware application developers;
- Quality of Context (QoC) modeling to specify quality levels when context information is obtained from imperfect sensors in the environment;
- Context management middleware that facilitates context information acquisition, reasoning, discovery, and distribution;
- Reference context-aware service platforms that uses middleware solutions to provide context-aware services.

### 2.1.1 Context Information Modeling

Abstract and concrete context

Context can be defined at an abstract level [39], without taking into account its digital concrete representation that is handled by computer systems. This level is useful for establishing what context is relevant in the user-application interaction under consideration, while abstracting from

the way in which the application will be able to capture, process, and use this context. An abstract context model helps designers of context-aware applications to understand better and early in the development process the application domain without the added complexity of considering technology dependent aspects. The abstract context model can be specified and later refined in the application implementation phase to consider the technology specific issues.

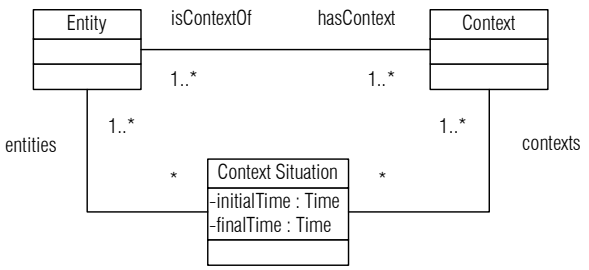
Abstract context definition

We adopt in this thesis the abstract definition of context from Merriam-Webster [64]: *Context is the set of, possibly interrelated, conditions in which an entity exists*. This definition refers to the general concept of context as a real-world phenomenon, and not to the digital representation of this phenomenon in computer systems.

Context modeling approaches

The context modeling approach of Dockhorn Costa [39] provides conceptual foundations that can be extended and specialized with specific concerns. These foundations include the (meta-)concepts of *Entity*, *Context*, and *Context Situation*, depicted in Figure *2-1*. An *Entity* is associated with instances of Context and Context Situations. A *Context* instance has a value modeled as a data type, a timestamp indicating the moment in time the context information value was determined. A *Context Situation* has a duration, which is defined by the moment the situation begins to hold (initial time) and the time the situation ceases to hold (final time), and a value that is also modeled as a data type.

Figure *2-1* Context information model




Comparison of context modeling approaches

The context information model of Dockhorn Costa has a broader scope for context modeling with respect to temporal aspects in comparison to the context modeling approaches of Henricksen & Indulska [55] and Chen [23]. These approaches, in contrast to Dockhorn Costa's approach, do not consider a suitable notion of time. Therefore, temporal aspects such as context situations, duration and precedence of situations cannot be explicitly defined.

Context situations and events

The context information model of Dockhorn Costa supports temporal aspects through the concept of context situations, which represents changes of interest in a set of context information attributes. When changes in a context situation occur, situation events are generated when

the situation begins to hold (enter true situation event) and when the situation ceases to hold (enter false situation event). For example, it is possible to model a context situation in which a person stays within a certain distance to another given person for a certain period, or a situation in which a patient has had a body temperature above a certain threshold during the past day. Context information events that are not related to context situations are called primitive events.

Context situation

A *Context Situation* is defined as "*a particular state-of-affairs that is of interest to applications. A situation is a composite concept whose constituents may be (a combination of) entities and their context conditions*". A *Context Situation* is a composite class that captures situations of interest that can be used by any context information consumer, including a context-aware service provider[1]. A context situation allows the definition of concepts like *close by*, which models the situation in which a person stays physically close at a certain pre-defined distance (e.g., five meters) from another person during a certain period of time. The context situation *close by* can be reused in a composite context situation that expresses that a person has been close to another one more than three times during one day. *Context Situations* are defined graphically at a higher level of abstraction using UML class diagrams enriched with constraints defined using the Object Constraint Language (OCL) to capture the particular state of affairs of interest.

Context-aware service developers can use a context information model to specify the concepts of interest for their context-aware service domain. For instance, developers of context-aware health applications might specialize a context information model to support the specification of an entity called *patient*, a context type called *body temperature*, and a context situation called *Fever*. Context information can be of different types and can also be acquired using different technologies. Examples of context information instances and context providers are:

- The speed and location of a user retrieved from the user's portable GPS device;
- The availability and presence (online, off-line) of a user retrieved from the user's instant communication application (e.g., MSN messenger or Skype);
- The physical activity (walking, running, cycling, or sitting) of a user inferred from accelerometers and sensors placed in the user's devices and environment;
- The schedule of a user retrieved from the user's calendar application (e.g., MS Outlook).

[1] In this thesis, we refer to *Service Provider* as always meaning a *Context-Aware Service Provider*.

In order to represent the abstract concept of context in a computer system, we adopt and extend in this thesis the context model of Dockhorn Costa. We extend this model with digital identities in Chapter 3 and we use the concept of *Context Situations* in the specification of context-based policies in Chapter 4 of this thesis. We chose Dockhorn Costa's model because of its expressiveness and also because it was available through an open source implementation that could be specialized and used by us in our case studies in Chapter 3 and 4.

### 2.1.2 Quality of Context Modeling

One of the first papers about Quality of Context (QoC) was written by Buchholz et al. [18]. They define QoC as "*any information that describes the quality of information that is used as context information*" and complement their definition by saying that "*QoC refers to information and not to the process or the hardware component that possibly provides the information*". The importance of modeling QoC is detailed in the work of Buchholz et al. [18] and Sheikh et al. [113]. QoC modeling is important for specification of privacy preferences, to improve the adaptation capability of context-aware services, and to allow the establishment of QoC level agreements.

The specification of QoC levels is useful for defining privacy preferences, by providing lower quality levels of the context information in the source that produces this information. The assumption is that context information of lower quality is less privacy-sensitive because less detail about the user environment is exposed to receivers of the information. Therefore, lower quality context information implies the information is less privacy sensitive.

Quality-of-Context representation is important to improve the adaptation capability of context-aware services. Context-aware service providers benefit more from context information if they received high quality information and are informed about the quality level of the information they receive. It is important for the service providers to know the quality level to allow them to make adaptation decisions. High quality context information increases the service provider capabilities of adapting reliably their services to the context of the service users.

Quality-of-context specification is important to allow the establishment of QoC level agreements. Context providers and consumers need a way of specifying contracts with respect to provisioning of context information, for example, a service provider to correctly adapt its services might require the almost real time provisioning of GPS location information of the service users. QoC agreements can be defined based on specification of minimum QoC requirements for correct service operation.

Table *2-1* presents the QoC attributes defined by Buchholz et al. and Table *2-2* describes the QoC attributes defined by Sheikh et al., which are a specialization of Buchholz et al. Buchholz et al. position trustworthiness as one of the QoC attributes and states that *trustworthiness is used by the context provider to rate the quality of the actor from which the context provider originally received the context information*. However, their definition of QoC contradicts with the definition of trustworthiness because they explicitly state that QoC is about the information and not the process nor hardware component that provides the information.

Table *2-1* QoC attributes proposed by Buchholz et al. [18]

| Attribute name | Definition |
|---|---|
| Precision | how exactly the provided context information mirrors the reality |
| Probability of Correctness | the probability that a piece of context information is correct |
| Trustworthiness | how likely it is that the provided information is correct |
| Up-to-dateness | the age of context information |
| Resolution | the granularity of information |

Table *2-2* QoC attributes proposed by Sheikh et al. [113]

| Attribute name | Definition |
|---|---|
| Precision | the granularity with which context information describes a real-world situation |
| Probability of Correctness | the probability that an instance of context accurately represents the corresponding real world situation, as assessed by the context source, at the time it was determined |
| Trustworthiness | mentioned but not defined, out of their scope |
| Freshness | the time that elapses between the determination of context information and its delivery to a requester |
| Spatial Resolution | the precision with which the physical area to which an instance of context information is applicable is expressed |
| Temporal Resolution | the period of time to which a single instance of context information is applicable |

Sheikh et al. [113] specializes the QoC definition of Buchholz et al. by splitting the resolution quality attribute into two different types: temporal and spatial resolution. These two resolution operations specify the granularity of the context information provided with respect to the timestamp information and with respect to the physical space the context information refers to. According to Sheikh et al., their proposed quality attributes do not apply to all types of context and are only relevant for context information about physical entities. For example, the spatial resolution attribute does not apply to the context information of a person's body temperature.

Sheikh et al. go one step further than the other QoC models because they not only propose a QoC model but also quantification strategies for the quality attributes and show how QoC can be used for privacy protection. Sheikh et al. do not develop further the concept of trustworthiness because it is out of the scope of their work.

Huebscher et al. [60] propose a QoC model and an algorithm to rank context providers according to their QoC capabilities using QoC attributes, including a QoC attribute related to the trustworthiness of the context information. They represent QoC levels as points in a multidimensional space where each dimension represents the measurement of one QoC attribute. For example, considering only precision (x) and refresh rate (y) the QoC level is a point in a two dimensional space (x,y).

Using a multidimensional point representation for QoC levels Huebscher et al. proposes to compare the different QoC levels by computing the Euclidean distance as a ranking metric for context providers. A context information consumer could specify its requirements using a point, and is able to select the best context provider that satisfies the specified requirements by computing all the euclidean distances to the available providers and select the one with the shorter distance. Their QoC model does not present details about the quantification of the QoC attributes, nor do they specifically mention which set of quality attributes should be included in their model. Huebscher et al. [61] also propose a learning model for QoC trustworthiness that calculates the trustworthiness values based on binary positive/negative feedback from the users.

### 2.1.3 Context Management Middleware

In this section we discuss existing middleware solutions for context management. We start by presenting an overview of the functionalities found in middleware solutions for context-aware systems in general followed by a detailed description of two approaches: the Context Handling Platform (CHP) and the Context Management Framework (CMF). We chose to describe in detail these two approaches because they are adopted by us and are part of the AWARENESS research project [119], which also includes the work developed in this thesis. The focus of the AWARENESS research project is precisely an *infrastructure that enables rapid and easy development of context-aware and pro-active applications in a secure and privacy-conscious manner*.

#### *Functionalities*

According to two surveys conducted by Chen & Kotz [22] and Baldauf et al. [7], existing context-aware middleware solutions address the following aspects: *sensing infrastructure*, *context modeling*, *context processing*, *historical context data*, *resource discovery*, and *security and privacy*.

The sensing infrastructure consists of hardware (sensors) and software components distributed in the environment to capture context information. These components provide an interface to access the context information, which is represented using a context information model. There is no standard interface or context information model and each of the existing middleware solutions adopt their own specific solution. Examples are Semantic Web approaches using ontologies for context modeling and Web Services interfaces to retrieve context information from the sensing infrastructure[2].

Context information retrieved from the sensing infrastructure is in most cases of low level of abstraction. An example of low level context is the list of the physical addresses of all Bluetooth devices in the vicinity of a Bluetooth dongle. This low level information is not particularly useful for context-aware adaptation of services because the service user can not be identified and the location information is not explicitly declared. However, this information can be further processed using additional information to infer that a person, the device owner, is located in a position nearby the current physical location of the Bluetooth dongle. Processing of context information includes also advanced reasoning techniques using rule engines and knowledge bases that are capable of inferring higher level information based on the low level information retrieved from the sensing infrastructure. Context processing approaches also include support to historical context data, which allows prediction of context information considering past patterns and learning algorithms.

The discovery of resources is a generic functionality also found in general purpose middleware solutions like CORBA and Web Services. In the particular case of context-aware systems, resource discovery is tailored to the context-aware functionality. Example of tailored functionality is the discovery of context provisioning components capable of providing context about specific entities, Quality of Context (QoC) requirements, and at specific situations. In most cases, users of context-aware systems are mobile, which requires special resource discovery support to identify sensing infrastructure components when roaming in foreign domains.

Considering the aspects addressed in existing context-aware middleware, a common set of roles can be identified, namely: the *Context Provider*, the *Context Registry*, and the *Context Consumer*. The context provider acquires raw data from sensors in the environment and produces context information about a specific set of entities [3] at a certain quality level. The context type, supported QoC level, and the entities that the context pro-

[2] For further details about context modeling see Subsection 2.1.1

[3] In a context-aware service platform, context providers may not interact directly, only with sensors. They may also reason about and combine context information from other context providers. We consider this scenario outside the scope of this thesis.

vider can produce context about (a.k.a. *Context Owners*) are advertised to the context registry. When a context consumer needs context about a certain entity, it queries the context registry and may inform a minimum quality level in the discovery request. The context registry returns to the context consumer a list of context providers together with the supported QoC level and supported context owners.

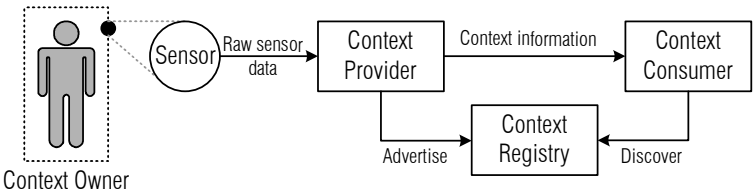



Figure *2-2* Context discovery and provisioning

In Figure *2-2*, the context registry is a registry of context providers with similar functionality of a service registry [121]. However, a context registry may also perform negotiations [17], in case the context owners' privacy preferences are considered in the discovery process. For a description of a negotiation strategy where the context provider QoC constraints, the QoC requirements of the context consumer, and the context owner's privacy constraints are taken into account in the discovery process, we refer to the work of Sheikh et al. [113].

After the discovery of the context providers (Figure *2-2*), the context consumer requests context about a specific context owner and receives the context information together with a reference to the QoC attributes associated to the context information provider. The QoC attributes advertised by the context providers may not correspond to the QoC level advertised by the context provider, because the supported QoC level may change dynamically according to the environment conditions. For this reason, the QoC attributes are always sent with the respective instances of the context information to indicate the current capabilities of the context provider.

Security and privacy is implemented in existing systems using policy languages to express security requirements about context information distribution and processing. This policies specify usage rules with respect to context information and security schemes for authentication and secure communication between the components that handle context information. Considering that trust and privacy are the focus of this thesis, we discuss these topics in detail in Sections 2.2 and 2.3.

***Context Handling Platform (CHP)***

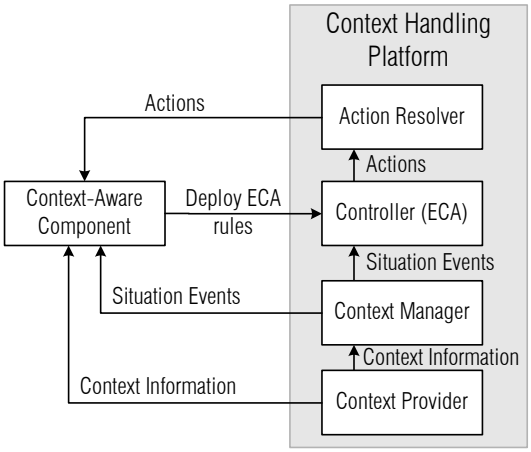
The Context Handling Platform (CHP), proposed by Dockhorn Costa [39], consists of a context model and a context management architecture. The context model used in the CHP was already introduced in Subsection

2.1.1, including the concepts of context information and context situations. The CHP introduces an architecture with context management components to realize the concepts specified in the context information model.

Context Providers, Context Managers, and Controllers

Context situations are realized in the CHP using a rule-based approach that allows the detection of context situations. In the architecture of the CHP (see Figure *2-3*), context information is captured by sensors in the environment and is accessed through the *Context Provider* components (a.k.a. *Context Sources*[4]). The *Context Manager* component uses the context information retrieved from the *Context Providers* to detect the *context situations* and to generate *context situation events* accordingly, when the *context situations* begin and end.

Figure *2-3* Context Handling Platform (CHP) architecture




The situation events are captured by a Controller component that implements Event-Condition-Action (ECA) rules and is responsible for triggering, for example, application adaptation actions. ECA rules are deployed in the *Controller* component by a *Context-Aware Component*, which is a component interested in adapting its behavior according to changes in context information or context situations. A *Context-Aware Component* might also deploy ECA rules to trigger specific actions, which are carried out by an *Action Resolver* component. By deploying ECA rules, the *Context-Aware Component* can delegate context-aware functionalities to the *Controller* component.

Architectural benefits of the *Context Handling Platform*

Developers of context-aware components use the CHP by subscribing to the *Context Providers* and *Context Managers*, respectively, to receive context information values and context situation event updates. Another possibility is to subscribe directly to the *Controller* component using an

[4] In the *Context Handling Platform*, the *Context Provider* components of this thesis are referred to as *Context Source*.

ECA rule that specifies occurrences of events, context information values, situation events, and actions. Rules in the Event-Condition-Action (ECA) format can be specified using a specialized ECA rule language for context-aware service platforms called ECA-DL [39]. The CHP has no support for sensing, context information discovery, and also does not implement security or privacy functionalities. The main focus of the CHP is on context modeling and processing.

***Context Management Framework (CMF)***

The Context Management Framework (CMF) [118] is a middleware solution developed by the Novay Research Institute. CMF supports context sensing, processing, discovery, and security and privacy aspects for interaction between administrative domains. Figure *2-4* depicts the CMF architecture. Each administrative domain running the CMF framework has an *User Manager*, *Context Providers*, and *Context Sources*.

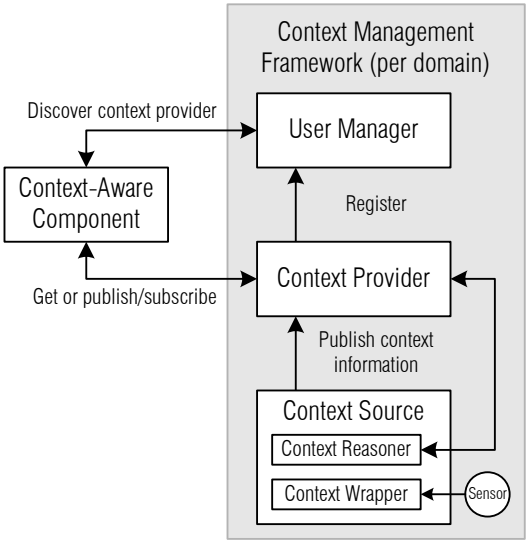



Figure *2-4* Context Management Framework (CMF) architecture

The context source component acquires and process sensor data from the environment, and makes it available to context providers, acting as a *Context Wrapper*. Context sources are also capable of combining and processing context information from other context sources acting as *Context Reasoners* and forming an hierarchy of context sources/providers. Context providers implement synchronous (get) and asynchronous (publish/subscribe) interfaces to allow *Context-Aware Components* access to context information. Context providers are capable of providing information about many users in the domain, and the *User Manager* is responsible for keeping track of all context providers associated to a specific user.

The security and privacy support in the CMF framework include authorization policies and privacy by design with respect to discovery of context providers. The CMF framework implements authorization policies specified by the system administrators [57] using XACML. The context provider discovery process has built in privacy when roaming users want to use the context sensing and processing infrastructure of a foreign domain [56].

### 2.1.4 Reference Context-Aware Service Platform

In this subsection, we describe the reference context-aware service platform we adopt in this thesis. We specified this reference context-aware service platform based on our experience in the AWARENESS research project [119].

***Roles and Interactions***

Figure *2-5* illustrates the five roles we distinguish, namely the *Service User*, *Context Owner*, *Identity Provider*, *Context Provider*, and *Service Provider*. The arrows in Figure *2-5* indicate the basic interactions between the roles when a user accesses a service provider. The box with a dotted line that surrounds the *Context Owner* represents sensors in the environment that collect context information about this entity.

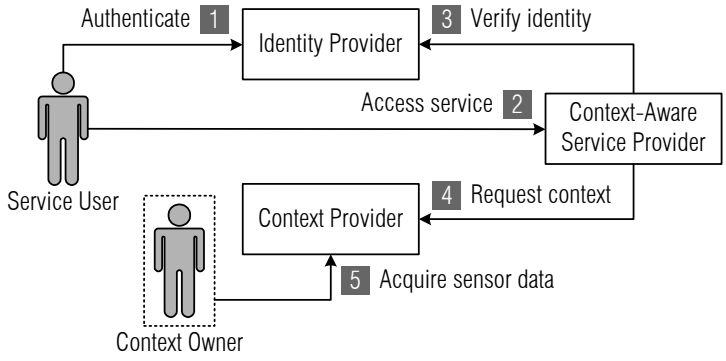



Figure *2-5* Roles in a context-aware service platform and their interactions when a user accesses a service provider

First, the *Service User* authenticates with the *Identity Provider* and receives an identity token (1). After the authentication is performed, the *Service User* requests access to a service provided by the *Service Provider* (2), which will verify the identity token of the user in order to grant access to the service (3). To be able to adapt the service to the relevant context, the *Service Provider* requests context information about the *Context Owner* from the *Context Provider* (5). This context information is retrieved by the *Context Provider* from sensors in the physical environment of the *Context Owner* and might be, for instance, the current activity or location of the *Context Owner*.

The list below presents the description of each of the roles we present:

- Service User: the entity that uses a context-aware service;
- Context Owner: the entity to which the context information refers;
- Context Provider: the entity that is capable of providing context information about another entity;
- Identity Provider: the entity that authenticates other entities' credentials;
- Service Provider: the entity that provides services customized to the context information of context owners relevant to the service being provided. Context-aware services perform context-based adaptation of the service provided.

Context owners and service users may be the same entity

In Figure *2-5* we show the service user and the context owner as different entities; however, they may be roles played by the same entity. We show these two entities as separate roles to emphasize that the service provider may use context information about other entities that are relevant for the context-aware service being provided other than the service user.

Even though Figure *2-5* presents (for reasons of simplicity) only user authentication and identity verification, with only one identity provider (arrows 1 and 3), the *Context Owner*, the *Service Provider*, the *Context Provider*, and the *Identity Provider* itself should also provide digital identities when interacting with other entities. Even so, we not expected for all the entities to be authenticated with the same identity provider component.

The service provider may use context information that references more than one context owner as well. In a particular context-aware service provisioning scenario, it is possible that multiple entities play the same role, and that one entity plays more than one role. For example, a person holding a GPS device may play, at the same time, the roles of the user, the context owner, and the context provider when accessing a service from a service provider that uses the location information retrieved from the GPS device.

***Example Context-Aware Service***

Figure *2-6* presents a fictitious example context-aware service. In this example, an user authenticates with an Identity Provider and receives an identity token. This identity token is used to access the navigation service that helps the user to find the directions to a destination based on the user's current location. The location of the user is retrieved from the user's mobile phone provider by the navigation service provider using the identity token. The mobile phone provider determines the user's location based on the GSM access point the user is currently connected to.

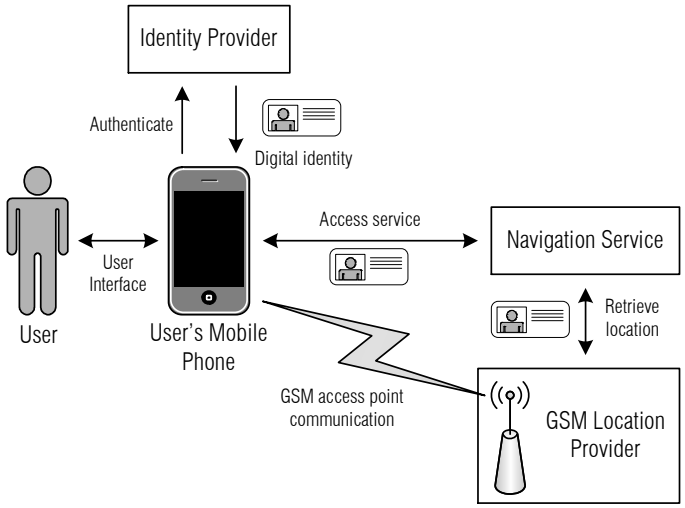



Figure *2-6* Example of context-aware service instance

## 2.2 Trust Management

In the context of computer security there is no consensus about the semantics of the terms *trust* and *trust management*. Different research communities use these terms with different meanings. According to Gollmann [46], the term *trust* is used formally and informally in existing computer security research to refer to a trusted computing base (TBC), trusted code, trust management approaches, security policies and mechanisms, and trusted computing technology.

Trust Definition

In this thesis, we adopt a definition of trust inspired by Abdul-Rahman & Hailes [2], Jøsang [68], and Quinn et al. [107]. We define trust as *the measurement of the belief from a trusting party point of view (trustor) with respect to a trusted party (trustee) focused on a specific trust aspect that possibly implies a benefit or a risk* [5]. This trust belief is represented as a trust relationship between a trustor and a trustee. For example, *Bob* (trustor) believes that *Alice* (trustee) is reliable to arrive on time for her appointments.

Trust Management

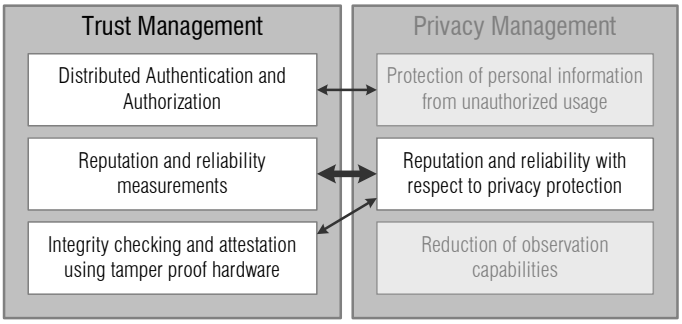
With respect to *trust management*, we adopt the definition from Jøsang et al. [74]: *trust management is the activity of creating systems and methods that allow relying parties to make assessments and decisions regarding the dependability of potential transactions involving risk, and that also allow players and system owners to increase and correctly represent the reliability of themselves and their systems*. Existing trust management approaches support the specification, bootstrapping, and evolution of trust relationships.

[5] Our trust definition and formalization are described in Section 3.3 of this thesis.

Figure *2-7* illustrate the relation between trust and privacy management[6] and our classification of the existing work in trust management in three main areas presented in this section:

1. *Distributed Authentication and Authorization*: trust values associated to identities and credentials are computed in a distributed system and used in combination with a set of rules to decide if actions requested by subjects should be allowed (positive authorization) or denied (negative authorization a.k.a. refrain authorization);
2. *Reputation and reliability measurements*: trust values focusing in specific aspects are computed based on direct experiences or indirectly through recommendations from third parties. These trust values are used to support the selection of entities, services, or products;
3. *Integrity checking and attestation using tamper proof hardware*: focus in trust guarantees using technical solutions that are theoretically or practically proven secure. Trusted computing chips such as the Trusted Platform Module (TPM) and smart card solutions are part of this research area.

Figure *2-7* Trust management research areas




These three main research areas are reflected in the structure of this subsection. In addition to these main areas, we start this subsection with related work on trust from a social perspective, and we explicitly discuss existing work in trust management that uses context information as an opportunity to enhance their functionalities in the end. Existing work in trust management focusing in reputation and reliability measurements with respect to privacy protection is also described in this subsection. Furthermore, existing work in privacy management that address authorization and access control that is not explicitly labeled as *trust management* is not described in this section. The trust management models we describe in this section were selected because they are seminal work (e.g. PolicyMaker), they are authoritative references in the area (e.g. RT and Sultan), or focus in scenarios connected to context-aware service scenarios (e.g. PTM and COMITY Framework).

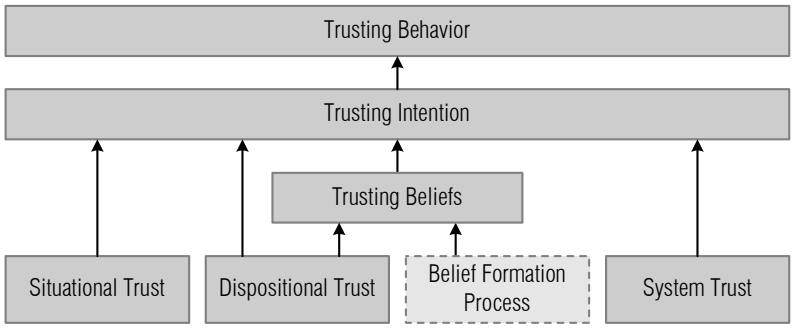
[6] the classification of the existing work in *privacy management* is detailed in Section 2.3

### 2.2.1 Social Trust

Six trust-related constructs

Mcknight & Chervany [88] identify six inter-related social trust constructs (see Figure *2-8*) based on a broad literature study. These constructs are: trusting behavior, trusting intention, trusting beliefs, situational trust, system trust, and dispositional trust. The main idea behind these constructs is that a set of **trusting beliefs** leads to a **trusting intention** that is manifested in a **trusting behavior**, or, in the authors' own words, "*when one has trusting beliefs about another, one is willing to depend on that person, then one will behave in ways that manifest that intention to depend*".

Figure *2-8* Relationship between the six social trust constructs




The constructs of trusting beliefs, trusting intention, and trusting behavior are summarized below:

- **Trusting beliefs**: is the extent to which one believes (and feels confident in believing) that the other person in a specific situation will function or react in a specific way. Trusting beliefs are person- and situation-specific and are influenced by the dispositional trust;
- **Trusting intention**: is the extent to which one is willing to depend on another person to perform a task on his/her behalf;
- **Trusting behavior**: is the extent to which one person voluntarily depends on another person in a specific situation with a feeling of relative security, even though negative consequences or harm are possible.

From the six trust constructs, Trusting Beliefs, Intention, and Behavior include both cognitive and affective components of trust. The cognitive component of trust is related to the rational subjective probability while the affective component is the pure human feeling of confidence. The most prevalent concepts related to the construct of trusting beliefs are clustered into four categories of behaviors: benevolence, honesty, competence, and predictability. For a detailed description of these categories and behaviors we refer to Mcknight & Chervany [88].

The *belief formation process* (see Figure *2-8*) is a subjective process related to the perception and feelings of the person and his/her previous

experiences. The outcome of this process is a subject belief to trust and is not formally defined. The construct of trusting beliefs is the most important but not the only decisive determinant of trusting intention and trusting behavior. Dispositional Trust, System Trust, and Situational Trust also influence the trusting intention and behavior, and are defined below:

- **Dispositional trust** (basic trust): means that a person can manifest trust intentions because of a pre-disposition to trust in a cross-situational, cross-personal way. Dispositional trust is the tendency of a person to trust across a broad spectrum of situations and persons. As illustrated in Figure *2-8*, the disposition to trust influences the trust belief and the trusting intention constructs [89]. Dispositional trust is, for instance, the trust an employee has in his/her new colleagues even though (s)he does not know any of them personally yet;
- **System trust** (impersonal trust): is a trust concept that does not consider personal attributes of the other, but rather the impersonal structural assurances and situation normality such as regulations, safeguards, and the perception that things appear normal. System trust could be described as the trust an employee has on his/her company taking into consideration that working laws and regulations are in place;
- **Situational trust**: means that one has formed an intention to trust a non-specific other party every time a particular situation arises. This is an individual situation strategy and is, for instance, the trust a person has in tourist activities when traveling to a specific country that has a good reputation with respect to the safety of its tourists.

Dispositional, System, and Situational Trust are types of trust. Dispositional Trust or "basic trust" describes the general trusting attitude of an entity. System Trust or impersonal trust refers to a trust that is not based on any property or state but rather on the perceived properties of or reliance on the system on which the trust exists. Situational Trust represents the extent to which one party intends to depend on a non-specific other party in a given situation.

In the trust model proposed by this thesis, we adopt the concepts of trusting beliefs and situational trust from Mcknight & Chervany and we consider the other types of trust out of scope. We relate the situational trust concept with our context-based policy management mechanism, which supports context-based trust management policies. Situational trust is important in context-aware services because the focus is also in the adaptation of the services to the situation of the service users. Furthermore, trusting beliefs is an important concept to support the selection of entities to interact with in this service scenario.

### 2.2.2 Authentication and Authorization

***Identity Management***

Existing research on identity management addresses the problem of digital identification in networked computer systems. This problem includes techniques to represent digital identities, verify identities, communicate digital identities to third parties, and models for identity-based access control. Authentication and access control are therefore one integral part of identity management solutions. Jøsang et al. [72] introduce four identity management architectures and describe the trust requirements in each of these architectures.

In the *isolated* identity architecture each administrative domain manages their own digital identities and no identity information is propagated to other domains. In the *federated* identity architecture each administrative domain manages their own digital identities and these different identities are used across different administrative domains. In *centralized* identity architectures one specific third party (a.k.a. identity provider) is responsible for the provisioning of digital identities and these identities are used in multiple administrative domains. Finally, in a *personal* authentication management architecture the identity holder him/herself manages his/her different identities and identities are not shared across different administrative domains. In the personal architecture Jøsang et al. suggest that users should manage their identities with help of a tamper-proof personal authentication device.

According to Jøsang et al. [72], the trust issues in the different architectures are related to many different aspects, for example, privacy protection, authentication mechanisms, care with respect to identity handling, correctness of identity mapping, compliance with data correlation policies, and tamper-resistance. Furthermore, the different identity architectures can be combined, and is is not clear how the trust issues and the relation between the different trust aspects in the resulting hybrid identity management architecture looks like. Furthermore, the *personal* identity architecture is positioned as the most flexible and low trust requirements because the identity holder does not depend in heavily on third parties to protect his/her privacy.

Liberty Alliance [106] and MSN Passport are examples of identity architectures to support Single-Sign-On (SSO). When using these approaches, credentials of an user accessing a service in one domain is communicated through assertions to an identity provider, which is responsible for verifying the authentication information. In spite of being aimed only at identification issues, identity management approaches are sometimes wrongly applied to other trust aspects. Some approaches assume that, if the identity of some entity is certified, this also means implicitly that the privacy policies or context information provided by this entity can be

trusted as well [84]. The study of Jøsang et al. [72] shows that in identity management architectures different trust aspects are relevant and should be explicitly addressed.

***PolicyMaker, Keynote, and REFEREE***

The term *Trust Management* was introduced by Blaze et al. in the PolicyMaker [16], KeyNote [15, 14], and REFEREE [25] trust management systems. Blaze et al. were the first to study trust management and to define a generic framework for policies and trust relationships that supports the specification of authorized actions based on the trustworthiness of credentials. Trust management is defined by Blaze et al. [16] as *a unified approach to specifying and interpreting security policies, credentials, and relationships that allows direct authorization of security-critical actions*.

In PolicyMaker, KeyNote, and REFEREE [43] policies specify that the right to execute an action should be granted to the holder of a specific public key. Prior to the work started by Blaze et al., authorization policies were specified in the applications, and there was no generic support for specification or reasoning about trust that could be used by application developers. The specification of trust decisions was hard coded in the application logic and could not be easily changed or adapted by the application developers or system administrators considering changes in the application's trust requirements.

***Role-based Trust Management (RT)***

Role-based Trust Management (RT) [31] is a family of languages ($RT_0$, $RT_1$, $RT_2$, $RT^T$, $RT^D$, and $RT_\ominus$) for distribute authorization and delegation. The main motivation behind RT is that an authorization decision in a distributed system depends on attributes of a principal that are in some cases unknown to the decision maker. For example, an web shop which decides to give a discount to the students of all universities member of the 3TU Federation in the Netherlands does not necessarily knows the member universities nor the enrolled students.

In RT, authorities are responsible for issuing statements in the form of credentials or certificates. These statements are maintained in a distributed manner and have their integrity ensured through cryptographic signatures. When an entity needs to take an authorization decision, these statements are analyzed using delegation rules to determined if an entity is member of a role. Authorization policies are defined by assigning permissions to a role. An example of a role definition is *WebShop.User*. In this example *User* is the role and *WebShop* is the role owner, which is the only entity authorized to determine the role members. The semantic of a role in $RT_0$ is simply of a set.

In $RT_0$ there are four types of credentials that can be issued by a principal: *Simple Member* (a), *Simple Inclusion* (b), *Linking Inclusion* (c), and *Intersection Inclusion* (d). Informally, the objective of each of this credentials is to assign principals to a role, to specify that role includes all members of another role, to specify delegation of role ownership, and to specify partial delegation. In the RT terminology a set of credentials issued by a principal is called a policy. To illustrate these different types of credentials we introduce an example with the following policy:

- *UniversityOfTwente.EnrolledStudent* ⟵ *Ricardo*: *UniversityOfTwente* states that *Ricardo* is part of the role *EnrolledStudent* (a);
- *UniversityOfTwente.Student* ⟵ *UniversityOfTwente.EnrolledStudent*: *UniversityOfTwente* states that all members of *EnrolledStudent* are members of the role *Student* (b);
- *3TUFederation.University* ⟵ *UniversityOfTwente*: *3TUFederation* states that *UniversityOfTwente* is a member of the *University* role (a);
- *WebShop.Student* ⟵ *3TUFederation.University.Student*: *WebShop* delegates the membership of *Student* role to members of the role *3TUFederation.University* (c);
- *WebShop.Discount* ⟵ *WebShop.Student* ∩ *WebShop.RegisteredUser*: *WebShop* states that members of *Discount* role are all the members of the role *Student* that are also members of the role *RegisteredUser* (d);

Permissions in RT are specified through the association of authorization semantics to roles. For example, *Webshop.AllowAccess* does not specify explicitly which access should be allowed. The meaning of the authorization policy must be expressed elsewhere and there is no implicit meaning associated with this role. For further examples of policies and credentials please consult Czenko et al. [31].

In addition to the formalization of credentials and policies $RT_0$ also specifies thr formalization of a credential chain discovery algorithm. This algorithm specify types of queries that are needed to be performed in order to determine, given a set of policies, all members of an specific role. Furthermore, RT also specifies a distribute credential storage system.

$RT_0$ is only one of the members of the RT family of languages. $RT_1$ extends $RT_0$ with parametrized roles to allow more flexibility in the definition of credentials. $RT_2$ extends $RT_1$ with logical objects that constrain the set of possible values occurring in credentials. $RT^T$ supports threshold and separation of duty policies. $RT^D$ supports delegation of role activations that can be used to temporarily delegate authority. Finally, $RT_\ominus$ adds non-monotonic extensions to the RT family of languages allowing negation of role memberships in a particular context.

***Sultan Framework***

Sultan is a high-level trust specification and analysis framework for Internet application providers developed by Grandinson & Sloman [50, 49] and is considered representative in this area by the trust management community. In the Sultan framework, trust management is defined as "*the activity of collecting, codifying, analyzing and presenting evidence relating to competence, honesty, security or dependability with the purpose of making assessments and decisions regarding trust relationships for Internet applications*". Evidence is related to identity and qualification proofs, risk assessments, user experience, or recommendations from third parties.

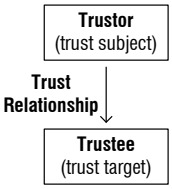



Figure *2-9* Trust relationship

The Sultan framework specifies trust following the same basic notion of trust relationship that has also been adopted by many other researchers [48]. Trust is broadly defined as a relationship between two entities, the *Trustor* and the *Trustee*. The subject that places trusts in a target is the *trustor*, and the target entity that is trusted is called the *Trustee* (Figure *2-9*).

In the Sultan framework, a *trust relationship* has associated with it a value, adheres to some action, and it has auxiliary properties that influence it. For example, it is possible to specify a trust relationship using the following abstract syntax: { trust (Trustor, Trustee, action(parameter), level ) ? property(Trustee) } , which means that *Trustor* trusts *Trustee* at level *level* to perform *action* if *?property* of the *Trustee* is equal to true. A concrete trust relationship example stating that the system administrator (*superuser*) trusts the user *neisse* at the maximum trust level (*100*) to access files at his home directory when the user is not on holidays is { trust (superuser, neisse, open(homeFolder), 100) ? NotOnHolidays(neisse) }

The Sultan framework makes an explicit distinction between trust and authorization. For example, if a trust relationship states that an entity is trusted to perform network security tests, this does not necessarily mean that this entity will be given unrestricted access to the network. In the Sultan framework, authorization statements have to be made explicit, taking into account the specified trust relationships. The authorization statements are considered a refinement of the trust relationship specifications.

The Sultan framework is divided into four major components available to the system administrator: the specification editor, the analysis tool, the risk service, and the monitoring service. The specification editor is a user interface for trust specification, storage, and translation. The basic primitives for trust specification are trust, distrust, positive recommendation and negative recommendation.

Trust and distrust statements are in the format "*Trustor trust/distrusts Trustee at Level if Constraint is true*". Levels of trust are specified between -100 and 100, where distrust is quantified using zero or negative numbers and and trust is represented with positive numbers. Positive and negative

recommendations are in the format "*Recommender recommends/does not recommend Recommendee at recommendation Level to perform Action if Constraint is true*". After the initial trust specifications are entered, the analysis tool allows querying using logic programming statements in Prolog. Queries can be specified to find conflicting trust relationships and recommendations, for example, to find a set of trust relationships that lead to conflicting authorization decisions.

The system administrator has to manually define queries to analyze specific scenarios, for instance, to verify if a certain action is allowed, or to verify if there are conflicting trust specifications. The analysis tool is used together with the risk service to support decision-making and conflict analysis, and to allow calculation of risk levels. The concept of Risk is beyond the scope of this thesis and for this reason is not detailed here. The monitoring service provides an interface for users to provide feedback and generates notifications for the system administrator if the user feedback results in conflicts and ambiguities. The monitoring service does not provide output to the users of the system; it works simply as a unidirectional communication channel.

***Pervasive Trust Model (PTM)***

The Pervasive Trust Model (PTM) [3] is a decentralized trust management model for pervasive computing environments where each device is responsible for managing its trust relationships with other entities. The basic idea behind the PTM model is to provide support for ad-hoc trust relationships, recommendations from third parties, and trust evolution based on user feedback. The PTM trust model is targeted at constrained portable devices and for this reason has been designed with simplicity in mind. The simplicity of the model is demonstrated by its limited functionality, which allows computation of trust values based only on the trustworthiness of entities' certificates and uses a simple algorithm to exchange and combine trust recommendations.

PTM adopts a trust definition from Jøsang [68], which is similar to the definition we adopt and defines trust as "*the belief that an entity has about [an] other entity, from past experiences, knowledge about the [latter] entity's nature and/or recommendations from trusted entities. This belief expresses an expectation on the entity's behavior, which implies a risk.*" In our trust definition we state that trust implies not only a risk but also a benefit. The PTM model supports two types of trust relationships: direct trust by previous knowledge and indirect trust based on recommendations. The final indirect trust degree for an entity is calculated as the average of the recommendations and only recommendations from trusted identities are processed. The calculated trust degrees are used by PTM to specify access control policies, allowing access to information owned by trustors to trustees with a trust value above a certain threshold.

***COMITY Framework***

The COMITY framework developed by Corradi et al. [28] is a middleware solution for pervasive systems that supports the adaptation of trust relationships based on the context conditions. The main idea of COMITY is to associate context attributes with trust degrees, in the sense that trustees will be associated with trust degrees if their context information satisfies a set of context conditions. Trust degrees are then associated with authorization and negative authorization (a.k.a. refrain) policies specifying which actions are allowed and not allowed to be performed for trustees that acquire a certain trust degree. The dynamic association of trust degrees with context conditions allows, as a result, context-dependent management of trust and authorization decisions.

The context model adopted by COMITY differentiates context into physical and logical context. The physical context type references physical spaces delimited by geographical coordinates. The logical context type references other types of context, for example, user role, user activity, temporal conditions, and resource availability. Trust is represented as levels in the set {very trustworthy, trustworthy, untrustworthy, very untrustworthy}.

### 2.2.3 Reliability and Reputation

***Reputation in Peer-to-Peer Systems***

Many decentralized trust models to assess reputation of entities in Peer-to-Peer (P2P) systems have been proposed. Authoritative models in this area are Poblano [24] for Java JXTA [65], and Eigentrust [77] for general purpose data distribution P2P systems.

The main objective of the Poblano trust model is to assess trustworthiness of nodes' certificates and trust values to rank and select nodes that are good sources of information for specific keywords. Eigentrust has a similar objective, however, a unique trust value is computed and assigned to each node reflecting the combined experience of all other nodes in the P2P network. This combined trust values indicates that a node has not lied to its own benefit or provided inappropriate material to other nodes in the network.

***Privacy Protection Reliability***

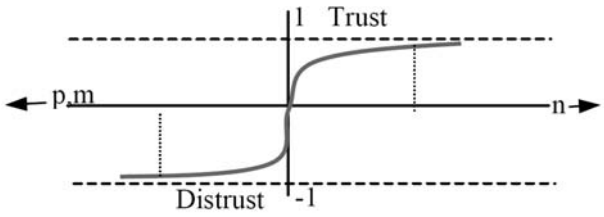
Daskapan et al. [34] propose a trust model that specifically addresses trust aspects related to the privacy of context-aware service users. Their approach provides a heuristic model to evaluate the trustworthiness of context consumers in order to influence user privacy policy decisions. If the evaluated trust of the context consumer is under a certain pre-defined threshold, then user consent is required to authorize access to the context information. If the evaluated trust is above a certain threshold then the

context provider decides, on behalf of the user, to authorize access to the context information.

The resulting trust value of a context information consumer (trustee) in the trust model of Daskapan et al. is a function of the number of previous experiences of the service user (n), the distance between the trustor's trusted certification authority and the trustee's digital identity certification authority in number of hops (m), and the a priori probability of the trustor to distrust the trustee based on a given function (p).

Daskapan et al. define the trust function based on the assumption that trust grows gradually and has a maximum of full trust/distrust that is never reached. To model this trust behavior, they chose to use an adapted *arctangent* function (see Figure *2-10*). They chose the *arctangent* function because this function starts from the origin and increases exponentially over time never reaching an upper limit, which is the same behavior the authors believe trust values should assume over time because it is impossible to be completely sure about the trustworthiness of an assumption (there is always uncertainty).

Figure *2-10*
Arctangent function




The mechanism proposed by Kolari et al. [79, 78] enhances the P3P framework[7] to include trust values associated with the websites. The enhanced P3P framework proposed by Kolari et al. uses the REI policy language as the implementation choice to specify the privacy preferences in the enhanced P3P framework and the concept of trust is modeled using a reference ontology. One instance of the trust ontology concept is presented in Figure *2-11*.

In their trust ontology, the trust value depends on many aspects, including the popularity of the website calculated from the number of references to the website (e.g., retrieved from Google PageRank [47]), the presence of a P3P policy, and the domain name extension. Kolari et al. do not detail in their work how precisely the concepts of this ontology influence the resulting trust values or how the trust calculations are made.

***Reputation from End-user Point of View***

Reputation mechanisms are implemented as a user-centered approach for realability trust evaluation, for example, in E-bay [40] and the Internet

[7] P3P is introduced in the following section.

Movie Database (IMDb) [35]. These reputation mechanisms focus on the feedback users give about content provided by the applications or about other users of the systems with respect to specific trust aspects, such as reputation as a buyer, reputation as a seller, or personal taste regarding a film. The objective of these reputation mechanisms is to support users in selecting content or other users to interact with.

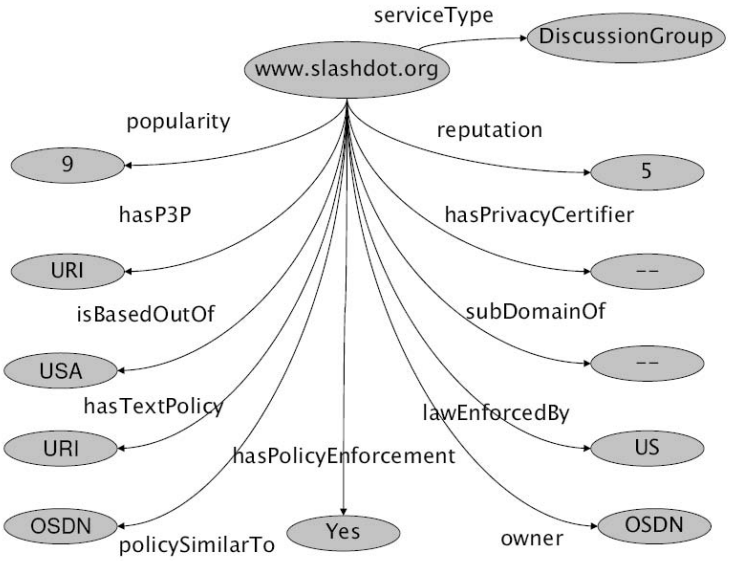



Figure *2-11* Trust-enhanced web privacy framework

### 2.2.4 Context-based Trust Management

Toivonen et al. [114, 115] propose using context information as input for trust evaluation to improve existing trust management frameworks. These proposals are based on the observation that the context situation influences trust decisions in real-world scenarios. In these proposals, the inference of different levels of trustworthiness of a piece of data depends on the currently active context of the data provider. These approach is different than other trust management approaches (e.g. PolicyMaker and Sultan) [48] that associate a trust degree with an specific entity (trustee). Taking an entity-centric perspective it is possible to state with the same effect that the trust degree is associate with the entity for that specific moment in time when the data was provided.

Trustworthiness of context information instances is also addressed by Hulsebosch et al. [62]. In their approach, the trustworthiness of context information providers previously unknown is determined by comparing the context information values retrieved from these providers with the

context information values retrieved from trustworthy context providers. If the context information values of the unknown and the trustworthy context providers match, then the unknown context providers are considered trustworthy sources.

They have implemented and tested their approach in a scenario that compares the location context information provided by train travelers with the location and velocity of the train, which is the trustworthy source of context information. The final objective of their work is to use the result of the comparison as input for a less intrusive user authentication process because the train travelers can prove they are on the train without presenting their identity. The proof of identity is the correct location and speed information that can not be easily predicted by users that are not indeed traveling in the train.

### 2.2.5 Trusted Computing

In existing research on computer security, the term trust is also used to refer to hardware secure tamper proof solutions using the Trusted Platform Module (TPM) chip proposed by the Trusted Computing Group (TCG) [51, 52, 53, 54]. In TPM enabled hosts, the trusted computing support is disabled by default, and in order to activate it the ownership of the chip must be taken. The TPM chip must be enabled in the computer BIOS and then the ownership can be set using a configuration password. After the ownership is set the TPM generates a set of owner-associated keys that are bound to a read-only protected key stored in the TPM memory called the Endorsement Key (EK).

The EK is written in the TPM chip at manufacturer time and is associated to the storage keys and integrity measurements. All the keys generated by the TPM chip are protected using the EK and are not stored in the TPM chip due to memory limitations. Nonetheless, the owner generated keys are protected because they are encrypted using the EK key that never leaves the TPM chip.

TCG's Core Roots of Trust

The TCG standard defines Core Roots of Trust for Measurement (CRTM), Storage (RTS) and Reporting (RTR). The core roots of trust are entities that provide a bootstrapping of trust and should be trusted by default. The CRTM is responsible for measuring the first pieces of code that are executed when a system is started. Common implementations of the CRTM are the BIOS or the CPU. The Root of Trust for Storage (RTS) and the Root of Trust for Reporting (RTR) are implemented in the Trust Platform Module (TPM) chip, which is a fundamental component in the TCG architecture. The RTS serves as a root of trust to tamper proof storage of encryption keys for software components running in the TPM enabled host.

The RTR is a root of trust root for remote reporting of integrity measurements (hash codes) computed by the CRTM. In practice, the RTR is mapped to an asymmetric RSA keys stored in the TPM chip that is used to sign the hashes of the hardware and software components computed by the CRTM. The hash codes are also stored in the TPM protected memory in registers called the Platform Configuration Registers (PCRs). The PCRs store SHA1 hash values in the TPM tamper protected internal memory.

The RTS and RTR are both supported by the hardware secure storage, and cryptographic operations using the keys and cryptographic processors implemented in the TPM chip. The roots of trust for storage and reporting depend on the manufacturer of the TPM chip that must ensure the secrecy of the keys written down to the internal and protected TPM memory at manufacturing time. If the internal keys are exposed then these roots of trust are compromised.

In the trusted boot process the Core Root of Trust for Measurement (CRTM) is the first code executed (see Figure *2-12*). The CRTM hashes the operating system loader code and the boot configuration parameters (dashed lines) before passing control to the operating system loader (solid lines). The operating system loader code must be trusted and will pass control to the operating system. Examples of measurements performed are hashes of the OS kernel, the services that are initialized, security policy enforcement mechanisms daemons, and the files containing the deployed security policies.

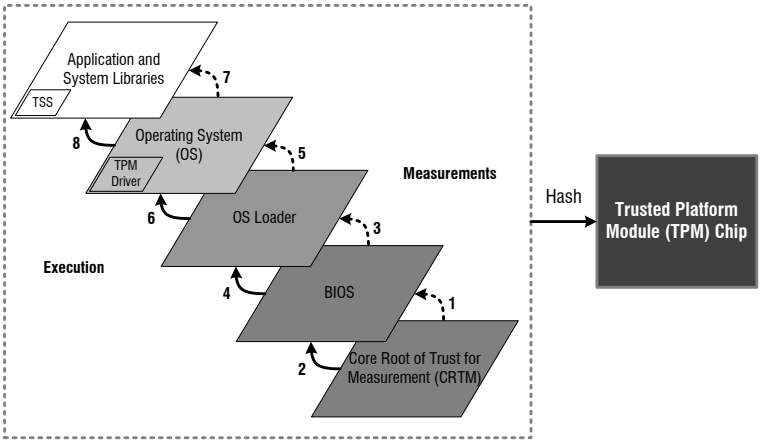



Figure *2-12* The trusted boot process

As depicted in Figure *2-12*, integrity checking is implemented at different levels of abstraction, for example, the BIOS code, the operating system loader, the operating system kernel and libraries, up to the application level [95]. The dependency between the different levels is called the chain of trust. The result of the trusted boot is the record of all

components executed in the chain of trust, which is stored in the tamper proof hardware and securely reported to remote entities through the remote attestation process. Trusted computing is also used to certify the identity of remote platforms and to strength identity management architectures [124].

## 2.3 Privacy Management

Privacy Definition

From a social perspective, privacy can be defined as *the quality or state of being apart from company or observation, freedom from unauthorized intrusion* [64]. According to this definition, full privacy could only be achieved by a solitary person who is completely unobservable and in total control of who is authorized to accompany and observe his/her life. Except for few cases where people choose to live in complete isolation [80], the social nature of human behavior makes full privacy impossible to achieve. People are then left with the other two options, namely, to maximize control over who is authorized to observer their life and to reduce the observation capabilities of others.

From an information technology point of view, privacy issues are restricted to digital access to information about people. In particular, context-aware services have a huge impact in users' privacy because sensors embedded in the environment constantly capture and store detailed information about the service users'. In this thesis we focus in privacy aspects from the information technology point of view, more specifically we focus in usage control aspects of users' context-information when using context-aware services. In context-aware services, the perceived privacy in the system impacts the user acceptance [66].

Figure *2-13* illustrate the relation between trust and privacy management and our classification of the existing work in privacy management in two areas that are introduced in this section:

1. *Protection of personal information from unauthorized usage*: this area is concerned with specification and enforcement of privacy preferences focusing in data usage control specification and enforcement;
2. *Reduction of observation capabilities*: solutions are concerned with obfuscation and anonymization of information to enable privacy in information systems by design.

### 2.3.1 Usage Control Specification and Enforcement

Specification and enforcing of usage control policies can be achieved using computer security models. Examples of classical formal security models are Bell-La Padula Model [8], Biba Integrity Model [13], and Clark-Wilson Model [26]. These models describe rules to ensure access control,

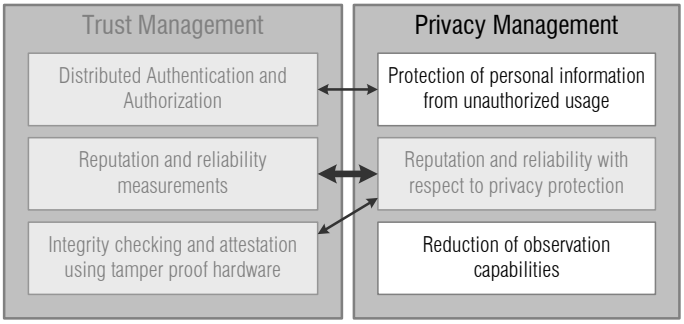



Figure *2-13* Privacy management research areas

integrity, and data confidentiality when subjects access data objects. With respect to access control, two classical approaches have been proposed [1]: Discretionary Access Control (DAC) and Mandatory Access Control (MAC). In DAC access rules are defined based on the identity or group to which a subject belongs and delegation of access permissions to other subjects is allowed. In MAC the system decides based on subject/object attributes if an access attempt should be allowed or denied and subjects are not allowed to delegate their permissions.

Newer alternative approaches to MAC and DAC are Lattice-based Access Control (LBAC) [36, 111], Role-based Access Control (RBAC) [112], and Attribute-Based Access Control (ABAC) [87]. The main trigger for these different approaches is the operational difficulty to express some security requirements using the conventional MAC and DAC approaches. Existing studies show that these models cam simulate each other, for example, RBAC can simulate MAC and DAC [38]. Therefore, the choice of security model is a matter of operational convenience depending on the application domain and security requirements.

Policy language implementations and policy frameworks are the concrete operation instances of security models. The eXtensible Access Control Markup Language (XACML) [45], for example, is a language that implements ABAC. Policy-based management approaches have their roots in the network management area, and allow the specification of part of the system functionalities using rules. The specification using rules makes the maintenance and configuration tasks more manageable because rules can be easily replaced or adapted without modifications in the system implementation [91]. Policy-based approaches also reduce the complexity in large systems because they allow reuse and modularization [102].

Usage control extends the problem of access control to the specification of future constraints in usage of data. Access control policies are restricted to the specification of provisions and authorization rules while usage control policies also include the specification of obligations that must be fulfilled after the data is released [11, 12]. Examples of policy models,

languages, and frameworks that support the specification of usage control policies are the UCON model [103], the Obligation Specification Language (OSL) [58, 105], and the Ponder2 framework [33, 32, 110, 117].

The Platform for Privacy Preferences (P3P) [30] is a World Wide Web Consortium (W3C) protocol that specifies the P3P policy language for websites to describe their privacy practices. The Enterprise Privacy Authorization Language (EPAL) [6] is the user counterpart of P3P for users to describe their privacy preferences. Users can configure their web browsers using EPAL to block access or receive a warning when the privacy practices of the website they are accessing specified in P3P do not match their privacy preferences.

The P3P framework only provides policies languages and no guarantees or trust evidence for users with with respect to the enforcement of the privacy practices presented by the websites. In other words, users have no guarantees and no support to trust the websites' stated privacy practices. Furthermore, P3P and EPAL do not define enforcement support, they only provide a conceptual language framework for specification of usage control policies and user requirements. Zuidweg et al. [125] propose to use P3P in a web services-based context-aware application platform.

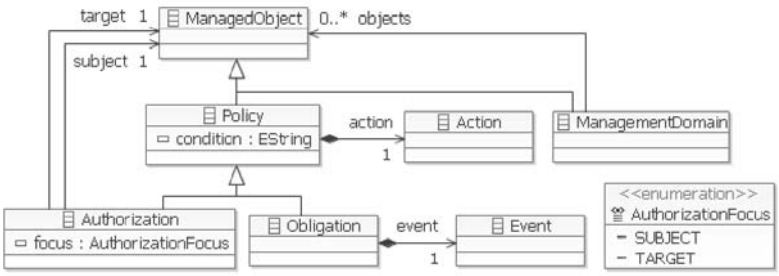
All existing security models and policy-based approaches have a basic set of shared concepts (e.g. authorizations and obligations) that can be understood by a detailed description of one of these approaches. Therefore, we describe in this subsection the Ponder2 policy framework, which is an established and well-known academic policy framework for usage control specification and enforcement. We use Ponder2 and XACML in our prototyping efforts for context-based policy specification and enforcement in Chapter 4. We chose Ponder2 because it supports authorizations and obligations, and was the only language with public available implementation that could be extended by us. We also describe in this section policy languages that support specification of context-base policies.

***Ponder2 Framework***

The Ponder2 framework is a general purpose policy-based solution for specification and enforcement of authorizations and obligations. The Ponder2 framework policy information model is depicted in the diagram in Figure *2-14*. The main concept in the Ponder2 policy language are *Managed Objects* that represent all the entities in an administrative domain that can be managed and controlled using the framework. Managed objects include application objects that communicate through message passing, *Management Domains*, amd *Policies*. The execution language for the Ponder2 framework is called *Ponder Talk*.

The current Ponder2 distribution is implemented in the Java programming language, and the application managed objects are mapped to Java object instances. Policies are specified to control the invocation of methods in the Java objects annotated with Ponder2 managed object keywords. Interfaces and components are also defined and implemented to support remote enforcement of policies on Java objects running in other virtual machines. This interfaces also include Java enforcement wrappers that could be used to specify and enforce policies on Web Service invocations implemented in Java.

Figure *2-14* Ponder2 Policy Information Model




The concept of a *Management Domain* is similar to a hierarchical directory structure, and helps in the policy management process because *Policies* can be specified for a set of *Managed Objects* instead of individual *Managed Objects*. The concept of *Management Domain* contributes to reduce the management complexity in large systems because it is impractical to specify and apply policies individually for each entity on a large scale. Using the management domain concept policies can be specified considering groups of managed objects.

Figure *2-15* Ponder2 management domain example

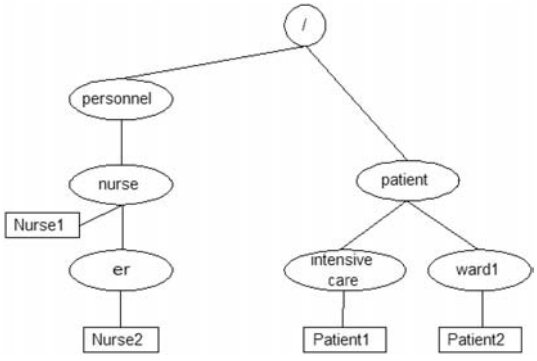



Figure *2-15* shows an example domain specification for a health scenario. In this example the root domain (/) contains two sub-domains called *personnel* and *patient*. The *personnel* domain is further divided into *nurse* domain that contains the emergency room(*er*) sub-domain. The

managed object *Nurse1* and *Nurse2* are respectively members of the *nurse* and *er* sub-domains. The *patient* domain is divided into *intensive care* and *ward1* sub-domains, which contain respectively the managed objects *Patient1* and *Patient2*. Policies can be specified and associated with specific managed objects or to a particular management domain, meaning that the policy applies for all direct and indirect members of the domain.

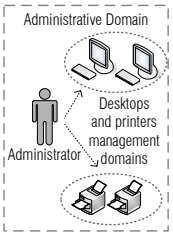



Figure *2-16* Administrative versus Management Domains

*Management Domains* are static sets, and the inclusion and removal of entities from a management domain has to be done manually. Management domains should not be confused with administrative domains, which are groups of resources under a common administrative entity (see Figure *2-16*) . One administrative domain might contain a set of management domain specifications in order to simplify the policy management task inside an administrative domain. In the Ponder2 terminology an administrative domain is referred as a *Self Managed Cell*.

Regarding policy types, Ponder2 supports the definition of authorization and obligation policies. Authorization policies specify actions that subjects are allowed to perform on targets. Obligation policies describe actions that subjects are obliged to perform when a specific event occurs. Both authorizations and obligations also include a condition specifying constraints on the enforcement, for example, checking the time of the day. The following listing presents an example of an obligation policy specified using Ponder2.

*Listing 2-1* Example Ponder2 obligation policy

```
policy
  event: root/event
  condition: [ :value | value > 100 ];
  action: [ :monitor :value | root print: "Value in event is " + value ];}
```

Obligation policies are specified using and Event-Condition-Action (ECA) rule pattern. In the previous example, the obligation rule is triggered whenever the event *root/event* is signaled. The *condition* part evaluates to true if the *value* attribute of the event is bigger then 100. Whenever the event is observed and the condition is evaluated to true the action part is executed. The action represents the obligation to be fulfilled. In this example the obligation is simply to print a message, however, more complex obligations with a sequence of actions can be specified.

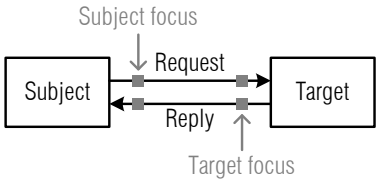
*Listing 2-2* shows an example of an authorization policy specified using Ponder2. This policy specifies that whenever the subject *root/personnel/nurse1* executes the action *getrecord* in the target *root/patient1* this should be allowed. The authorization policy also specifies the focus of the enforcement, which in this case is the target. The authorization focus indicate the enforcement point, which can be in the subject (s) when the action is triggered or in the target (t) when the action is received for execution. Figure *2-17* illustrates this difference between the two possible focuses of enforcement. This distinction allows more expressiveness in

the specification of policies because actions can be denied before they reach the target to avoid, for example, a Denial of Service (DoS) attack.

*Listing 2-2* Example Ponder2 authorization policy

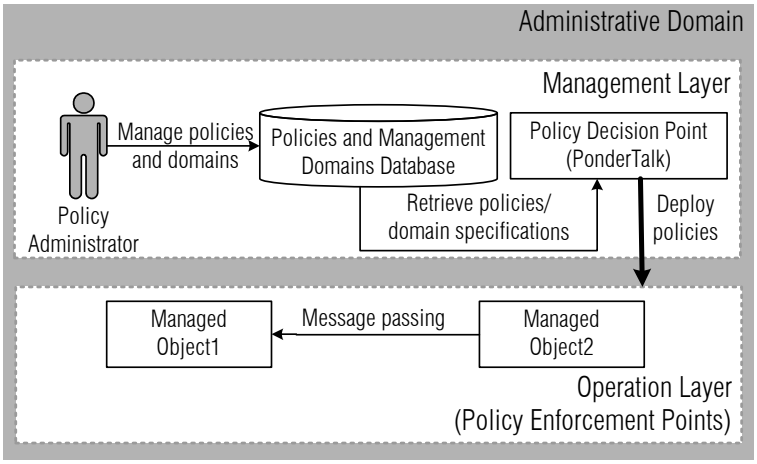
```
newauthpol subject: root/personnel/nurse1
  action: "getrecord"
  target: root/patient1
  focus:"t").
```

Figure *2-17* Target versus subject focus enforcement




In Ponder2, an administrative domain (a.k.a. Self Managed Cell) is logically divided in two layers: a *Management Layer* and an *Operation Layer* (see Figure *2-18*). Within the *Management Layer*, a *Policy Administrator* specifies the *Management Domains* and *Policies*, and stores them in a *Policies and Management Domains Database*. The *Management Domains* and *Policies* are then retrieved by a *Policy Decision Point* (PDP) that evaluates the policies with respect to the components in the *Operation Layer*. The specific policy enforcement logic implemented in the components of the *Operation Layer* are referred as *Policy Enforcement Points* (PEPs).

Figure *2-18* Ponder2 management and operation layers




With respect to its expressiveness, Ponder2 is not as powerful as the Obligation Specification Language (OSL) language or UCON policy model. Both UCON and OSL are based on Linear Temporal Logic (LTL) and support conditions that consider temporal ordering and cardinality of events. Ponder2 supports only simple propositional operators in the condition

part of an ECA rule. In contrast to Ponder2, at the time this thesis was written there was no public implementation of UCON or OSL available. A policy framework implementation using the OSL language was further developed by the author of this thesis [96].

***Context-based Policy Specification***
Joshi et al. [75] have extended the Role Based Access Control (RBAC) standard to support the definition of parameterized access control roles. Their proposal is called X-RBAC and provides dynamic management of access control roles based on time and location constraints. Their focus is specifically on access control policies for XML document sources at different levels (conceptual, schema, XML instance, and element).

The UbiCOSM framework [27] was developed by the same authors as the COMITY framework (see Subsection 2.1.4). The main objective of the UbiCOSM is to support context-based access control for ubiquitous service provisioning. The UbiCOSM framework uses the same context model as the COMITY framework. The basic difference is that the context conditions are not associated with trust degrees, but rather with access control lists. The approach proposed by UbiCOSM is very similar to the approach proposed by Covington et al. [29].

### 2.3.2 Reduction of Observation Capabilities

The work developed by Anciaux et al. [4] addresses privacy management for databases that store context information. In their approach, context information is stored in a database together with a privacy policy that describes the information life-cycle. The life-cycle policy (LCP) describes how the quality degradation of the information should take place in different quality dimensions, which are modeled as *context states*. In their approach, context information is modeled as a triplet, composed of the identity of the entity to which the instance of context information refers (a.k.a. context owner), the time when the context information was acquired, and the context information value. Each of these context triplets can have a different context state. Figure *2-19* [4] presents the set of context states for context of the type location. For each state, the values of the triplet assume different accuracy levels. A life-cycle policy (LCP) defines the initial state and the transitions for the context information instances. The authors prove that the solution is feasible and show the performance impact through a prototype and workload evaluations with and without the presence of LCPs.

Quality of context with a focus on privacy protection has also been addressed by Sheikh et al. [113]. Sheikh et al. go one step further than the other QoC models because they not only propose a QoC model but also quantification strategies for the quality attributes and show how QoC can

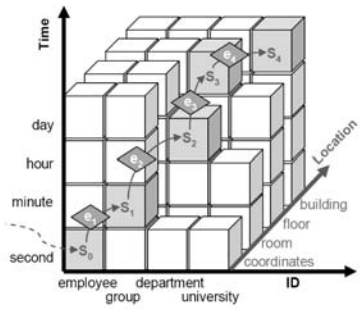



Figure *2-19* Cubic representation of location context states

be used for privacy protection [8]. Reduction of observation capabilities is also the focus of the k-Anonymity approach developed by Bettini et al. [10]. In the k-Anonymity work users are identified through pseudonyms and their location information is always communicated to service providers with a reduced quality. This type of privacy preserving technique using pseudonyms and identity anonymization is not the focus of the contributions in this thesis.

Hesselman et al. introduces an approach for privacy-aware discovery of context information providers [56]. In their solution, they introduce a *context agent* that is responsible for managing context information about a specific entity that roam across different administrative domains. Each entity's *context agent* is responsible for enforcing authorization policies for the entity's context information and to anonymize the identity of the entities in foreign domains. Following this approach, entities are able to benefit from context providers managed by foreign domains and remain anonymous to protect their privacy when roaming.

## 2.4 Summary and Discussion

In a context-aware service platform the provisioning of context-aware services depends on the collaboration of different service providers, namely context-aware service providers, identity providers, and context providers. When these service providers collaborate, trust relationships related to different trust aspects should be addressed considering the specific service functionality. For example, identity providers should be trusted with respect to identity provisioning services while context providers should be trusted with respect to context provisioning services.

While trust management support related to the provisioning of identities [3] and authorization roles [31] is already addressed in the litera-

[8] For a more complete overview of QoC modeling approaches, we refer the reader to Subsection 2.1.2

ture, trust management support considering trust aspects related to the provisioning of context information is not properly supported. Existing approaches for trustworthiness evaluation of context providers propose to address trust as an attribute of the context information, and position trustworthiness as one of the Quality of Context (QoC) attributes. From a trust management perspective we believe this is not adequate because in existing trust management approaches [48] trust is a degree associated to service or data providers and not to the data itself. The QoC model described by Buchholz et al. [18] suggest associating a trustworthiness value to the context information provider but they do not concretely support trust issues in their QoC model. Furthermore, the complexity in existing QoC models makes it difficult for developers of context-aware services to adopt these models and to understand precisely the meaning of the different QoC attributes.

In addition to the inadequate support for trustworthiness assessment of context information providers, existing trust management solutions do not support trustworthiness assessment when different trust aspects are combined. This is necessary a a context-aware service platform because when assessing the trustworthiness of a context-aware service provider, a service user should also consider the trustworthiness of the context providers and identity providers that cooperate in the service provisioning. Furthermore, the trustworthiness assessment depends on the choice of service users with respect to the trade-off between privacy and context-based service adaptation. For example, when users favor context-based adaptation instead of privacy the trustworthiness of context providers to provide context information is more important then the trustworthiness of context providers to protect the users' privacy. Existing trust management solutions focus on at most one trust aspect (e.g. identity [3], roles [31], or privacy [108, 34, 83, 78, 79]), and do not provide support for trustworthiness assessment when more trust aspects are combined or when user goals (e.g. privacy or context-based adaptation) should be considered in order to tailor the trust assessment strategy. We are not aware of any existing work describing a systematic analysis of the different trust aspects in context-aware service platforms.

Finally, in a context-aware platform users have different privacy and trust preferences depending on the situation they are in. For example[9], an user may decide to authorize access to their context information by a doctor in an emergency situation if this information is deleted by the doctor after the emergency situation is over. In addition to privacy, users may also define trust assignment strategies that depend o their situation. An example is an user that trust a set of service providers at home and do not trust this set of providers in his/her work environment. Exist-

[9] See Chapter 5 of this thesis for more examples

ing privacy management approaches focus on context-based authorization policies for context information [82, 62, 75, 29] and do not consider context-based obligations nor context-based trust policies. One possibility is to add context-based support to one general purpose policy language that supports obligations and trust policies, however, to the best of our knowledge this context-based support is not present in any of existing general purpose languages (e.g. Ponder2 [116] or OSL [58]).

In summary the following shortcomings were identified by us in the existing literature and are the focus of the contributions of this thesis:

- there is no adequate trust management support focusing on trust aspects related to the provisioning of context information and clear relation between trust and QoC attributes;
- existing trust management support does not consider user goals and combination of trust focusing on different trust aspects;
- there is no systematic analysis of trust issues in a context-aware service platform;
- existing privacy management solutions do not support context-based obligations;
- exiting trust management solutions do not support context-based trust policies.

Chapter 3

# Trust-based Selection of Context and Service Providers

This chapter [1] proposes a trust management model and mechanisms to support trust-based selection of context information providers and context-aware service providers. The objective of these mechanisms is to assist context-aware service users and service providers in managing the trade-off between privacy protection and context-aware service adaptation.

This chapter is further organized as follows. Section 3.1 describes an analysis of the trust relationships between the roles in a context-aware service platform. Section 3.2 describes our simplified QoC model that is used in the mechanism for trust-based selection of context providers. Section 3.3 introduces our abstract trust management model. Section 3.4 describes our mechanism for selection of context providers. Section 3.5 describes our mechanism for selection of service providers. Section 3.6 describes our two case study implementations where we apply our mechanisms in simulated context-aware service scenarios. Section 3.7 concludes this chapter with a summary of the contributions and final considerations.

## 3.1 Trust Relationships in a Context-Aware Service Platform

Trust management is necessary in context-aware service platforms because users and service providers, which are expected to be pervasive and numerous, need to judge whether new, existing, and previously unknown

[1] Parts of this chapter have been published in papers [98], [99], and [100] which were co-authored by the author of this thesis.

entities are (un)trustworthy to interact with. In this thesis we focus in trust issues related to identity provisioning, privacy enforcement, context information provisioning, and context-aware service provisioning. Figure *3-1* presents a summary of the trust relationships between the roles in a context-aware service platform we address in this thesis.

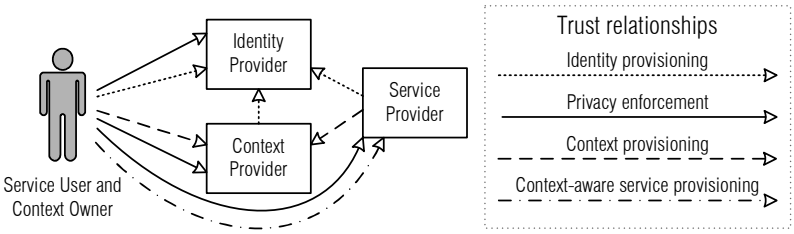



Figure *3-1* Trust relationships in a context-aware service platform

In Figure *3-1* the *Service User* and *Context Owner* roles are assigned to the same entity meaning that the service adapts its behavior to the context of the service user only. We acknowledge that this may not always be the case. For the a detailed discussion about the different roles and interactions between the respective roles we refer the reader to Figure *2-5*, Subsection 2.1.4.

The *Service User* should trust the *Service Provider* to reliably provide a specific context-aware service. The *Context Owner* should trust the *Context Provider* and the *Service Provider* to handle his/her context information. The *Context Owner* expects that his/her context information is released only when his/her privacy preferences authorize the access, and (s)he can only accept his/her context information to be communicated if (s)he trusts that both the *Context Provider* and the *Service Provider* are able and willing to adhere to his/her privacy preferences.

The *Service User* and the *Service Provider* should trust the *Context Provider* with respect to the provisioning of context information. This is important in order to guarantee that the information is provided with the required quality characteristics and consequently can be used in the expected context-aware service adaptation. Trust in the *Context Provider* from the *Service Provider*'s point of view is also required in case dynamic context-based security solutions requiring trustworthy context information are in place. One example is a service provider that only authorizes access to a service if the service user is at a specific location, for example, inside of an office building.

All the entities should trust the *Identity Provider* because it is responsible for the authentication and verification of credentials to allow the entities to identify themselves to other entities in the service platform.

The trust management model we describe allows the specification of trust relationships targeting the trust aspects depicted in Figure *3-1*. Each trust relationships focus on a specific trust aspect depending on the func-

tionality provided by the role. The set of trust relationships we address in Figure *3-1* is by no means exhaustive. Other trust relationships targeting different aspects may be required in other service scenarios depending on the functionality provided by the roles. Our trust management model proposes a basic set of trust aspects, based on our reference context-aware service platform, and motivates the definition of trust relationships focusing on these trust aspects. Our trust management model also includes trust assessment support considering the dependencies between these trust aspects.

For each of the trust relationships presented in Figure *3-1* it is possible to establish a trust value according to a certain aspect-specific metric. The following subsections present a detailed analysis of the aspect-specific trust relationships we identify and shows example trust metrics from the literature for obtaining trust values related to the set of trust aspects we consider. We also refine further the trust relationships giving more details about the trust relationships specified in Figure *3-1*. Our objective with this discussion is to motivate our trust management model and trust-based selection mechanisms described in Section 3.3.

### 3.1.1 Trustworthiness of Context Providers

Figure *3-2* shows the interaction and trust relationships between the *Context Owner* and *Context Provider* roles. The *Context Owner* registers his/her identity with the *Context Provider* and informs the *Context Provider* of his/her privacy preferences. The *Context Provider* stores the *(*Context Owner)'s privacy preferences and is expected to enforce the privacy preferences when the context information about the *Context Owner* is released to third parties (authorization enforcement), or to delete, reduce the quality, or anonymize the context information [85] after a certain period of time (obligation enforcement).

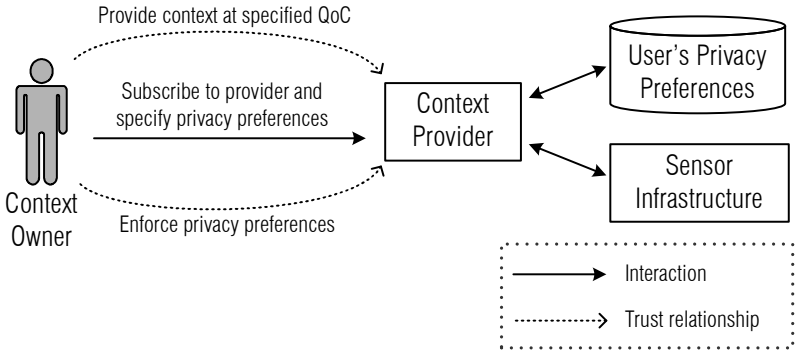



Figure *3-2* Trust relationships between user and context provider

We identify two types of trust relationships between the *Context Owner* and the *Context Provider*. The *Provide context at specified QoC level* trust re-

lationship is related to the reliability and competence of the *Context Provider* to provide context information according to a Quality of Context (QoC) level. The *Enforce privacy preferences* trust relationship is related to the competence and honesty of the *Context Provider* to enforce the *(*Context Owner)'s privacy preferences. These two relationships are respectively a refinement of the *Context provisioning* and *Privacy enforcement* trust relationships in Figure *3-2*.

Trust relationships between context owners and context providers

Trust metrics for context information provisioning

One existing approach [82] to evaluate the trustworthiness of context information providers takes into account the cryptographic trustworthiness of the context provider's identity. This approach is not adequate because the fact that the identity of a context provider uses trustworthy cryptography has no relation to the capabilities of this context provider with respect to the provisioning of context information. Other approaches to evaluating the trustworthiness of a context provider propose using the following metrics and mechanisms: reputation of the context provider established by a community, statistical analysis of the context information [60], and aggregation of context information from redundant context providers in order to increase the trustworthiness [62].

In the Context Handling Platform (CHP) context information is also realized using the concept of context situations and situation events (see Subsection 2.1.3). Context situations values are a composite of context information instances and the same trustworthiness evaluation approaches applicable to context information can be used. To evaluate the trustworthiness of situation detection components (a.k.a. Context Managers) trust values associated to the honesty and competence to observe and report situation events can be defined. Examples of these trust aspects are:

- Competence to observe situation events: a context manager component is capable of detecting all situation events meaning that all changes in the situation conditions are observed and there is no relevant context change that is not observed;
- Honesty to report situation events: a context manager component reports all observed situation events, is not omitting the reporting of observed events, and is also not deliberately reporting non-observed (fake) events;
- Competence to report events: the quality of context information values reported in situation events conforms to a specification;

Trust metrics for privacy enforcement

Trust mechanisms for evaluating privacy enforcement trustworthiness take into account the existence of information handling privacy policies defined by the context provider (e.g., P3P policies) [79]. The assumption made by this approach is that the existence of these privacy policies alone already contributes positively to the trustworthiness of the context

provider. Furthermore, if the privacy policies defined by the context provider match the privacy requirements of the context owner, there are no guarantees that the privacy requirements will be followed. We believe that such assumptions can only be made if tamper proof auditing mechanisms (e.g. using TPM devices) are in place to verify the enforcement of the privacy policies. Trust with respect to privacy can be also increased if manual Electronic Data Processing (EDP) audits are conducted (contrary to technology audits) and if the stated privacy policies are bound to legal liability of the context or service providers in case privacy violations are observed.

The following metrics are proposed by Daskapan et al. [34] and Kolari et al. [79] to calculate trust values regarding privacy enforcement aspects: user interest in sharing the information, confidentiality level of the information, number of positive previous experiences with the information consumer, number of hops between a provider and a consumer, a priori probability of distrusting, and service popularity in search engines. The number of hops is related to identity certification and the chain of certificate authorities between the information source (provider) and the target of the information (consumer).

Indirect privacy enforcement trust values can also be obtained through trust recommendations received from trusted third parties specialized in privacy protection issues. Privacy protection organizations take care of privacy policies certification in the same way identities are certified today by certification authorities [104]. Privacy recommendations might be provided also by informal organizations such as virtual users' communities and consumer protection organizations.

### 3.1.2 Trustworthiness of Identity Providers

Figure *3-3* shows the interaction and trust relationships between the *Service User* and the *Identity Provider* roles. The *Service User* subscribes and registers his/her identity profile information with the *Identity Provider* and provides his/her privacy preferences. The *Identity Provider* delivers as a result a digital identity, which can later be verified cryptographically by anyone. The *Identity Provider* is supposed to enforce the privacy preferences when someone requests access to the identity profile information, and should release only the allowed information if any information at all[2].

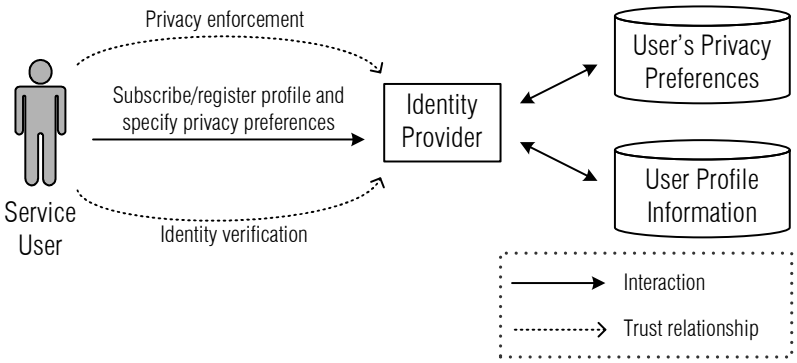
Trust relationships between identity holders and identity providers

We identify two types of trust relationships between an identity provider and an identity holder. The first trust relationship is related to the provisioning of identities that are reliable in the sense that they correctly identify an entity and the entity's profile attributes. The second trust relationship is related to the competence and honesty of the identity pro-

[2]One privacy preference might state that the identity should be completely anonymized.

vider in enforcing the identity holder's privacy preferences for the identity attributes.

Figure *3-3* Trust in identity provider




Trust metrics for identity provisioning

One metric that influences trust in the identity is the authentication method used. Identity providers that use very strong authentication, e.g. using SmartCard technology, can be relied on more to securely authenticate someone than identity providers that use only username/password authentication. The user registration policy also influences the identity provisioning trust. Identity providers that allow users to freely register without verifying the identity attributes of the user (e.g. Google and Yahoo) might not be trusted as much as identity providers that do not allow registration without doing some form of identity proofing, such as a university or a bank. With respect to privacy enforcement of identity attributes, similar techniques to those described for the context providers can be used.

### 3.1.3 Trustworthiness of Context-Aware Service Providers

Figure *3-4* shows the interactions and trust relationships between the *Service User* and the *Service Provider* roles. The *Service User* accesses the context-aware service provided by the *Service Provider* and includes in the access request his/her digital identity. The service provider verifies the service user's identity and requests context information about the *Service User* or other *(*Context Owners) relevant to the service being requested.

In Figure *3-4*, we assume that the *Service User* is also the *Context Owner* of interest for the service being provided. In some scenarios, a service invocation might not require the context information of the *Service User*. In this case, the privacy enforcement trust relationship does not apply, since the *(*Service User) is only concerned with the reliability of the context-aware adaptation. One example of this scenario could be a *Friend Finder* service that does not require the location information of the friend using the service. Only the location information of the target friend being located is relevant. However, users of a friend radar service will also be

targets of their friends and we expect that subscribers of context-aware services will, in most cases, also provide their context information for the service they will be using.

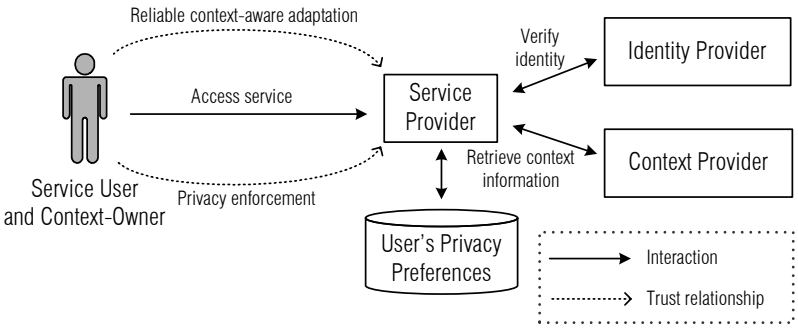



Figure *3-4* Trust in service provider

Trust relationships between service users and service providers

We identify two types of trust relationships between context owners, service users, and service providers. The first trust relationship is related to the enforcement of the privacy preferences of the *Context Owner* with respect to the context information that is being accessed by the *Service Provider*. Similar techniques to those described to evaluate the trustworthiness of context providers with respect to privacy enforcement can be used to assess trust values related to this trust relationship.

The second trust relationship is related to the competence of the *Service Provider* to reliably adapt to the context information. If the *Service Provider* is not competent, the result can be that the service provided is of less value. One example is a context-aware service that provides poor personalized tourist advice for a user in the sense that the advice is not correctly customized for the user's location.

With respect to the adaptation of the context-aware service, it is possible that the incorrect context-based adaptation is not the fault of the *Service Provider*. The *Service Provider* might be faulty due to untrustworthy context information retrieved from the context providers. Therefore, the context-aware service provider depends on his own competence to provide context-based adaptation and also depends on the reliability and competence of the *Context Provider* to provide context information about the context owners.

In this thesis we assume that the service provider is always competent, and that an unsuccessful context-based adaptation is related to context information provided by untrustworthy context providers. It is possible that a service provider does not reliably adapt despite receiving trustworthy context information at the required QoC level. This situation could be detected by analyzing the service implementation and the context information instances retrieved, and we consider it outside the scope of this thesis.

Table *3-1* Summary of trust aspects for each role

| Role ⇒ Trust Aspect ⇓ | Identity Provider | Context Provider | Service Provider |
|---|---|---|---|
| Identity provisioning (IDP) | ● | - | - |
| Privacy enforcement (PE) | ● | ● | ● |
| Context provisioning (CIP) | - | ● | - |
| Context-aware adaptation (CA) | - | - | ● |

Table *3-2* Summary of trust dependencies

| Role dependency ⇒ Role ⇓ | Identity Provider | Context Provider | Service Provider |
|---|---|---|---|
| Identity Provider | IDP | - | - |
| Context Provider | IDP | - | - |
| Service Provider | IDP | CIP | - |
| Service User | IDP | CIP | CA |
| Context Owner | IDP/PE | PE | PE |

### 3.1.4 Trust Aspects and Dependencies Summary

Table *3-1* and Table *3-2* summarize the roles, trust aspects, and trust dependencies we motivate. We do not present trust dependencies in the *Service User* and in the *Context Owner* roles because our initial set of trust aspects and mechanisms does not address any dependency on these roles. In this thesis, we focus on the trust aspects related to the trade-off between privacy and context-based service adaptation. An example of a dependency and trust aspect that could be considered, but is outside the scope of this thesis, is the dependency of the *Service Provider* on the *Service User* in the trust aspect of *paying for the service costs*.

In Table *3-1*, the trust aspect of *(*identity provisioning) applies to the *Identity Provider* role. The *privacy enforcement* trust aspect applies to the *Context Provider* and *Service Provider* roles, which manipulate the *(*Context Owner)'s context information. The *context provisioning* trust aspect applies only to the *Context Provider* role, and the *context-aware adaptation* (a.k.a. *context-based adaptation*) applies only to the *Service Provider* role.

In Table *3-2*, all roles including the *Identity Provider* itself depend on the *Identity Provider* role with respect to the *provisioning of identities* (IDP) aspect[3]. The *Service Provider* depends on the *Context Provider* with respect to the *provisioning of context information* (CIP) trust aspect. The *Service User* depends on the *Context Provider* with respect to the *provisioning of context information* (CIP) trust aspect. The *Context Owner* depends on the *Context Provider* with respect to the *privacy enforcement* (PE) trust aspect.

[3]We address this circular trust dependency in our trust management model.

## 3.2 Quality of Context Model

The following quality of context (QoC) attributes are identified as relevant in the literature [18, 60, 113]: *accuracy*, *precision*, *probability*, *probability of correctness*, *trustworthiness*, *temporal resolution*, *spatial resolution*, *up-to-dateness*, *freshness*, and *refresh rate*. Analyzing these attributes and the respective definitions, there is a clear agreement in the literature with respect to the set of quality attributes that are important; however, different names are given to the same quality concept. For instance, the concepts of accuracy, precision, and probability found in the literature are all measurements of precision.

Our QoC model relates the terminology of existing QoC models, based on an international standard for metrology [21], and models QoC as attributes of context information instances, with the exception of trustworthiness, which is defined as an attribute of the context provider [4] and is not considered a QoC attribute of the context information. There is no consensus in existing QoC models [113, 18] with respect to the inclusion of trustworthiness as one QoC aspect, and trustworthiness is not concretely supported by the any of the existing QoC models. Furthermore, existing QoC models do not describe precisely how trustworthiness relates to the other QoC concepts.

As already motivated in the previous subsection, in this thesis we support the proposition that trustworthiness is a property of the context provider from the context consumer's point of view, and is not a quality concept related to the context information itself [99]. Trustworthiness is related to the capability of the context provider to reliably describe the QoC levels he is capable of providing context information, and is not part of the quality concepts associated to the context information instances. Due to this different nature, we address trustworthiness in our trust management model as a trust aspect in the trust relationships with the context provider role in Section 3.3.2., and do not include trustworthiness in our QoC model.

We require a QoC model in this thesis because the trust management model we propose in Section 3.4.3 defines the trustworthiness level of context providers for a specific context information type with respect to a QoC level specification. For example, a location context provider might be trustworthy to provide location information with an error between 10 and 20 meters. This trustworthiness level is specified in our trust management model and is used in our trust management mechanism to support context consumers in selecting trustworthy context providers.

Existing QoC models specify QoC using quality attributes related to the context information instances to represent the quality aspects of the

[4] In this thesis, we refer to *context information providers* as *context providers*.

context information, and we follow the same approach in our QoC model. Our QoC model is based on existing QoC models (see Subsection 2.1.4) and contributes to the understanding of the existing QoC models. Existing QoC models define a variety of QoC attributes that overlap semantically, which makes it difficult to understand and choose a QoC model to adopt.

In our QoC model, we adopt as a reference the quality concepts and vocabulary proposed by the International Organization for Standardization (ISO) [21], and compare these standardized concepts with the QoC terms and concepts from existing QoC models in order to define our own QoC reference model. More specifically, we analyze and compare the quality concepts of accuracy, precision, and resolution from the ISO standard with the QoC attributes accuracy, precision, probability, spatial resolution, temporal resolution, and freshness from existing QoC models [18, 81, 60, 113].

***Accuracy and Precision***

The ISO standard defines accuracy as how close a measurement is to the real or to an accepted reference value, while precision is defined as how close together or how repeatable the results from a measurement are. Furthermore, according to the ISO definition [21], "*accuracy can not be expressed as a numeric value*", only inaccuracy can be measured as the error or percentage error [5]. For this reason, we do not consider accuracy and inaccuracy as relevant concepts in our QoC model. We assume that it is impossible for the context provider to determine for every context information request the real known value of the context information, so the inaccuracy can never be determined.

For illustration purposes, let us assume that the ambient temperature of a room is 25 degrees Celsius and a sensor in the room indicates 26 degrees; the error is then 1 degree Celsius. The inaccuracy information in this case is useful only for the calibration of context providers and for precision calculation when the real context information values are known, by repetitively comparing the context provider readings with the real known values in a controlled environment. It is impossible for a sensor to determine at every reading the real temperature value and provide a measurement of the inaccuracy together with the readings.

Using a graphical analogy, the repeated readings of a context provider can be related to shots at a target, where the center of the target (the bull's eye) is the true value of the context. Figure *3-5* represents the pattern of shots and how this would be interpreted as accuracy (low error) and precision of the context provider. Shots near the target center have a lower error and are considered more accurate while shots further away from the

[5] For simplicity reasons, we use the term accurate as meaning low error.

target's center are less accurate. Considering a sequence of shots, they are precise if they are all clustered together even tough they are inaccurate and not close to the center of the target.

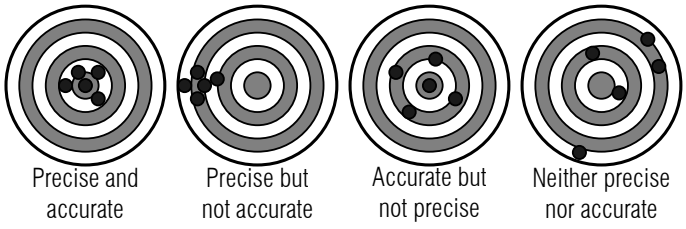



Figure *3-5* Target accuracy and precision

For numeric context information, precision can be expressed by means of significant digits, and in this case the average deviation is understood, even if not explicitly stated, as being one-half the value of the last significant digit. For example, the ambient temperature measurement of a room that records 25 degrees implies a variability of $\pm$ 0.5 degrees or 2% (0.5 divided by 25). In this case, the implicit precision is directly determined from the value representation of the context information, and may be used if no explicit precision information is defined. The standard measurements of probability, percentage difference, standard deviation, and variance can also be used to measure the precision of a context provider.

It is common for a context provider to reply with Boolean or discrete sets of values instead of numeric values, for instance, to determine if an entity is in a room or not. In case the context provider replies with Boolean values, the precision can be measured as the percentage proportion of true positive and true negative results in relation to the total number of results including the false positives and false negatives. For a presence sensor, for example, a precision of 100% indicates that the context provider always correctly identifies the presence/absence of an entity in a room. The ISO concept of precision overlaps with the concepts of precision and probability defined by Sheikh et al. [113] and Buchholz et al. [18] because both authors define precision and probability by means of repeatability of context information readings. Therefore, we believe that these concepts should be considered as a measurement of precision as recommended in the ISO standard for metrology [21] and should not be expressed as another QoC attribute with a different name.

***Quality Attributes Related to the Timestamp***

Existing QoC models define quality attributes related to the timestamp associated with the context information instance that are not related directly to the context information value. Examples of these attributes are freshness [113], up-to-dateness [18], and temporal resolution [113].

Freshness and up-to-dateness measure the age of the context information instance, from the moment it was determined by the context provider up to the time it is made available to the context information consumer.

*Temporal resolution* has been defined by Sheikh et al. [113] as *the period of time to which a single instance of context information is applicable or the best possible approximation of time at which a context was determined*. In our QoC model, we consider temporal resolution an implicit precision quality attribute of the timestamp associated with the context information, which can be measured by the number of significant time units available (e.g., year, month, day, hour, minute, etc.), or by a time duration, for example, by stating that the context information instance is valid for *2008/01/01 from 10PM until 12PM*. To allow a time resolution on the level of seconds, the context provider should include the *second* time unit in the timestamp of a context information instance.

***Precision of Location Information***

*Spatial resolution* has been defined by Sheikh et al. [113] as *the precision with which the physical area, to which an instance of context information is applicable, is expressed*. From the examples of spatial resolution presented by Sheikh et al. [113] we conclude that spatial resolution is only associated with the resolution of the location of a physical entity; therefore, we do not include spatial resolution as a separated QoC attribute in our QoC model. We adopt a more specific approach where temporal resolution is simply the *precision* of the *location* information.

***Analysis and QoC Model***

Table *3-3* summarizes our analysis of the QoC attributes from existing models and how they are related to the quality concepts of the ISO standard. Our conclusion is that the QoC models refer to accuracy, precision, and probability as always meaning the concept of precision with respect to the repeatability of measurements. Precision can be expressed by different means; however, we believe it is confusing to refer to the same concepts using different names. Therefore, we chose to adopt in our QoC model only the concept of precision without restricting the way that this precision attribute is measured (e.g., variance or standard deviation).

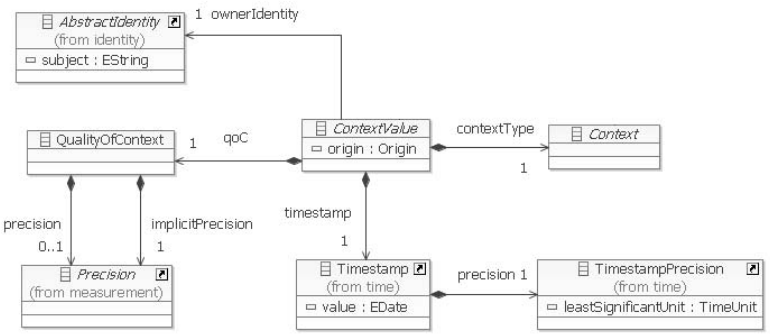
Figure *3-6* presents our QoC reference model, which is part of our reference context information model. In our QoC model, the *Context Value* instances are associated with a specific moment in time called *Timestamp*. Both *Context Value* and *Timestamp* have a precision measurement associated with them. The precision measurement covers all the definitions from the literature for precision, probability, and spatial resolution. The precision class is abstract, and depends on the context type; for example, numeric precision is represented in a different way than geographic location [63]

or timestamp precision. The implicit precision attribute is the precision measurement derived from the context value representation, which for numeric values is derived from the number of significant digits that have been used in the numeric representation.

Table *3-3* Mapping of QoC attributes to ISO concepts

| Quality Attribute | ISO Concept |
|---|---|
| Accuracy | Precision |
| Precision | Precision |
| Probability | Precision |
| Spatial Resolution | Precision of Location |
| Temporal Resolution | Precision of Timestamp |
| Freshness | - |

Figure *3-6* Quality of Context model




One example of a numeric context information instance could be a room temperature measurement with a value of *24.5 degrees Celsius*, with a precision value of $\pm$*1 degree Celsius*, and with a timestamp value of August 16, 2009, 23:30. The implicit precision can be calculated as $\pm$*0.5 degree Celsius* and the timestamp resolution as *minute*. Freshness can be calculated as the difference between the current time and the timestamp value.

In our QoC model, the implicit precision of the context information value, and the resolution and freshness of the timestamp, can be calculated from the context information value and the timestamp. For this reason, we do not explicitly include these concepts as classes in our model. Implicit precision and timestamp resolution are included in our model as methods of the *Timestamp* and *Context Value* classes. Our QoC model is generic, extensible, and supports QoC attributes related to the timestamp and context information values. Other QoC attributes can be defined focusing on other context types. The determination of the precision of the timestamp concerns issues related to clock synchronization and is beyond the scope of this thesis.

## 3.3 Trust Management Model

In this section, we present our trust management model, which supports the specification of aspect-specific trust relationships as identified in Section 3.1. Our trust management model instantiates well-known concepts like direct trust establishment through personal experience or beliefs, and indirect trust establishment through recommendations. Our trust management model quantifies trust using Subjective Logic (in short, SL) [69], which is a probabilistic logic capable of explicitly expressing uncertainty about the probability values.

We use our trust management model in two trust-based selection mechanisms to support users and service providers of context-aware services in managing their trust relationships and in selecting trustworthy entities to interact with. The first trust management mechanism we introduce supports service providers in selecting trustworthy context providers with respect to a QoC level. The second trust management mechanism supports service users in selecting context providers and service providers, taking into account the trade-off between privacy and context-based service adaptation.

Most of the existing trust management models (see Chapter 2 of this thesis) refer to a specific application domain and, as such, propose special-purpose solutions that are not easily portable to other domains. Our context-aware service platform domain requires a specific formalism of combining trust relationships focusing on specific trust aspects we have not found treated appropriately in the literature. For reasons of simplicity, we specify our trust formalism using a simple set of rules; however, we do not exclude that existing formalisms for trust (e.g. Nielsen & Krukow [101]) could be specialized to express and combine multiple trust aspects as required by our domain.

We formalize trust as a relationship between two entities, the Trustor and the Trustee, as widely accepted by the literature [48, 2, 68, 107]. In this thesis, we define trust as ***the measurement of the belief from a trusting party point of view (trustor) with respect to a trusted party (trustee) focused on a specific trust aspect that possibly implies a benefit or a risk***. For example, *Bob* (Trustor) may trust to a *high degree* (measurement) *Alice* (Trustee) concerning her *competence in coding in Java* (trust aspect). The risk implication is only present when Bob accepts to depend on Alice to code a Java program on his behalf or to use a Java code provided by Alice.

We refer to the term *trust* and *trust relationship* interchangeably, always meaning the relationship between two entities. We use the term *trustworthiness* to refer to the amount, measurement, or degree of trust in a trust relationship. Furthermore, the entities in a trust relationship that we refer to as Trustor and Trustee are digitally represented in an information system using digital identities.

The trust *aspect* models different scopes that can be tackled by the trust relationships. As motivated in Section 3.1, we address the following aspects of our reference context-aware service platform: identity provisioning, privacy enforcement, context information provisioning, and context-aware service provisioning. In Section 3.1 we also show examples of existing approaches for obtaining trust values for each of these aspects.

Trustors can perceive or interpret the trust aspects as an isolated or combined measurement of, for example, honesty, competency, reputation, usability, credibility, and reliability to perform a specific action. In this thesis, we assume that *part of the trust aspect*[6] always refers to a combination of the concepts of "honesty, competence, and reliability for a certain criteria", because this is the most common interpretation of trust observed by an extensive survey conducted by Mcknight and Chervany [88]. Other trust concepts are also important and are considered future work in this thesis. A list of possible trust concepts and their correlations based on user studies can also be found in [107].

Regarding the choice of the domain of *trust degrees*, existing trust management models have different proposals. Some authors quantify trust as a real numeric value (e.g., between $-1$ and $1$), a discrete value (e.g., trust or distrust), or a combination of both, where each element in the discrete set has a numeric equivalent (e.g., values in $(0, 1]$ mean trust, values in $[-1, 0)$ denote distrust, and $0$ means unknown). Our proposal is independent of any particular solution; we assume a generic domain $TValue$.

In this thesis, we instantiate $TValue$ in the set of opinions of the Subjective Logic (in short, SL) [69]. SL is a probabilistic logic capable of explicitly expressing uncertainty about the probability values. The basic assumption is that there is always uncertainty and that the truth is always expressed from an individual perspective. The SL formalism has been proven to be an appropriate formalism for addressing trust calculations because it allows more realistic modeling of real-world situations that reflect ignorance and uncertainty [73] compared to traditional probabilistic approaches [3]. In Subsection 3.4.2, we describe in detail the SL formalism, which is used in our trust mechanism to instantiate our $TValue$ domain.

With the expression

$$A \xrightarrow[v]{*;a} B$$

we indicate a trust relation between $A$ (the Trustor) and $B$ (the Trustee) that tackles the trust aspect $a$ and that has degree $v$.

[6] The trust aspect can be decomposed into the two parts: the goal and the criteria.

$B$ can also represent a category here. "$*$" is a placeholder for classes of trust relation. In this thesis, we consider two classes of trust relations: direct functional ($df$) and indirect functional ($if$) relations, so $* \in df, if$.

Direct trust originates from $A$'s direct experiences or evaluations of $B$. We distinguish two different sub-classes of direct trust: arbitrary and experience. Arbitrary trust is the trust determined based on personal beliefs without previous experience. Experience trust is trust determined based on $A$'s direct evidence that contribute to belief or disbelief.

Indirect trust originates when $A$'s resorts to indirectly evaluating $B$'s trust, for example, by combining trust values or asking for recommendations from other entities (see also [74]).

In our formalism, $A$ and $B$ are entities that belong to a set $ID$. Identities are assigned to different roles in different instances of our platform.

Aspect $a$ ranges over identity provisioning, privacy enforcement, and context information provisioning, that is $a = idp, pe, cip$. We consider the set of roles $R = \{US, CO, IP, CP, SP\}$ from our context-aware service platform, namely, user (US), context owner (CO), identity provider (IP), context provider (CP), and service provider (SP). The function role: $ID \rightarrow R$ returns the role that, at the present moment, a given entity identified by an identity $ID$ plays; initializing and updating this function is the exclusive competence of identity providers, but it can be invoked by any entity that has registered its identity.

We assume that entities can access a set of functions that calculate the direct trust value from a Trustor to a Trustee based on the evaluation of its privacy enforcement (pe), identity provisioning (idp), and context information provisioning (cip) qualities. These functions receive as input the Trustor and Trustee identities (ID ID) and return the trust value for the specific trust aspect:

$$trust\_PE : ID \times ID \rightarrow TValues$$
$$trust\_IDP : ID \times ID \rightarrow TValues$$
$$trust\_CIP : ID \times ID \rightarrow TValues$$

For example, $trust\_PE(Alice, Bob)$ is the evaluation of $Bob$'s honesty, competence, and reliability in its privacy enforcement aspect from Alice's point of view. Considering the metrics in Subsection 2.2, it is easy to image that Alice provides a trustworthiness profile against which Bob qualities are compared and evaluated. Here we assume a trusted third-party role, namely the Trust Provider whose task is to run those functions on demand and on behalf of the Trustors. These functions are our starting point for trust evaluation; on their output we can establish the degree of trust between the Trustor and the Trustee. If we specify our reasoning in term s of an inference system, i.e., in terms of axioms and deductive rules of the form premises/conclusion, the functions we have

identified in this section can be used, at a meta-level, to define our set of axioms. In all the following rules that express our algorithm, we assume that $role(A) = US$, that is, Trustor $A$ is a service user.

$$\frac{[trust\_PE(A,B)=v]}{A \xrightarrow[v]{df;pe} B} \quad role(B) \in \{CP, IP, SP\}$$

$$\frac{[trust\_IDP(A,B)=v]}{A \xrightarrow[v]{df;idp} B} \quad role(B) = IP$$

$$\frac{[trust\_CIP(A,B)=v]}{A \xrightarrow[v]{df;cip} B} \quad role(B) = CP$$

For example, in the first rule when $trust\_PE$ is invoked with parameters A and B, it returns a value v, which states that A has degree v of (direct) trust in B, with respect to the aspect pe (privacy enforcement). This aspect is significant when Trustee B is a context provider, an identity provider, and a service provider. In the following, we use the deductive style formalization to depict the main characteristic of our algorithm of trust evaluation and composition.

Figure *3-7* is our trust meta model showing how the concept of *Trust Belief* is related to the social trust concepts of *System Trust*, *Dispositional Trust*, and *Situational Trust* [86]. The concept of *Dispositional Trust* is the intrinsic/inherent disposition an entity has to trust any other given entity in the absence of evidence or previous experiences, which we believe can be used to support trust bootstrapping. The concept of *System Trust* is the impersonal trust perception an entity has regarding the set of regulations and safeguards of the system as a whole. For a complete description of all social trust types, consult Section 2.1.1 of this thesis.

A *Trust* relationship in our meta model is a class that references a trustor, and is composed of the degree of belief for the specific trust aspect. The *Situational Trust* represents the impersonal trust a *Trustor* has in one particular or all *Context Situations*, and the *Trust Belief* represents the trust a *Trustor* has in a specific *Trustee* and *Context Situation* with respect to a specific *Trust Aspect*.

In our trust management model we extended the context information model from Dockhorn Costa [39] (see Section 2.1) by adding an *identities* attribute to the *Entity* class. This attribute specifies that an entity may have one or more digital identities associated with it. The concept of identity is important in our trust management model to support identity-related trust issues and because entities may hold multiple digital identities. By identity we mean a digital identity that is issued and certified by a third party that we refer to in this thesis as an *Identity Provider*.

Figure *3-8* shows our identity meta model. In our identity meta model we support *Identities* issued by identity providers (issuer) and also the con-

cept of *Self Signed Identities*, which are identities that are hold and are certified by the identity provider itself.

Figure *3-7* Trust types model

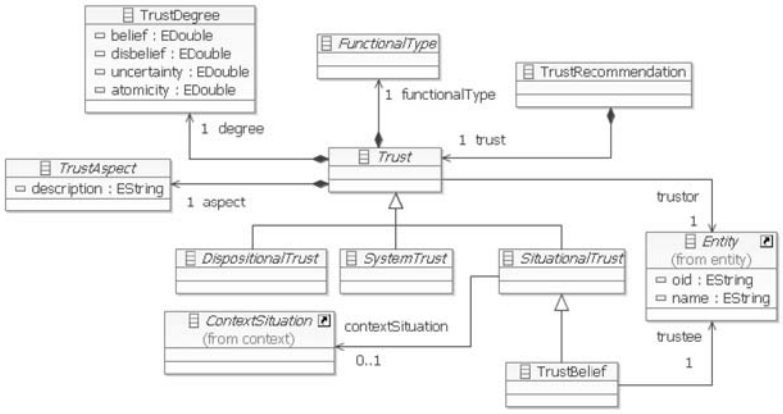
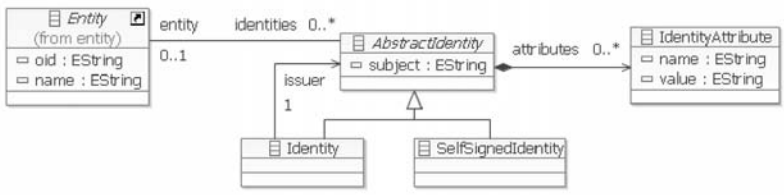



Figure *3-8* Identity model




***Subjective Logic***

In this thesis, we adopt the formalism of Subjective Logic (SL) [69, 73, 71] to quantify the degree of trust in a trust relationship specified in our trust management model. SL has been proven to allow a more realistic representation of real-world situations because it supports the expression of uncertainty regarding propositions. Traditional probabilistic approaches limit themselves to expressing belief and disbelief mass only. The strongest point of SL is that it combines the structure of binary logic with the capacity of probabilities to express the degrees of truth and uncertainty about propositions.

The basic assumption of SL is that nobody can ever determine whether a proposition about the world is true or false. Furthermore, the truth about a proposition is always expressed from the point of view of a specific individual, in the sense that it is subjective and unique to the person experiencing it and does not represent a general or objective point of view. In SL, the probability regarding a proposition is expressed through a Subjective Logic Opinion[7]. An Opinion is represented using the symbol: $\omega_x^A$,

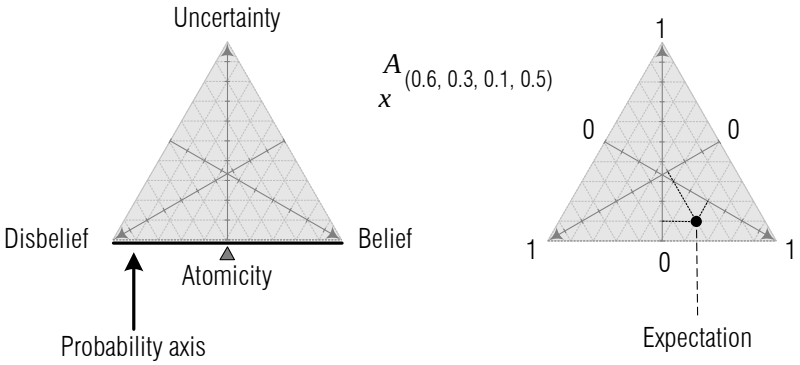
[7] We refer to *Opinion* from now on as meaning a *Subjective Logic Opinion*.

where $A$ is the belief owner and $x$ is the proposition. The opinion $\omega_x^A$ is a ordered quadruple $(b, d, u, a) \in [0, 1]^4$, where:

- $b$: is the belief mass supporting that the specified proposition is TRUE;
- $d$: is the belief mass supporting that the specified proposition is FALSE;
- $u$: is the uncertainty amount or uncommitted belief mass (neither TRUE or FALSE);
- $a$: is the base rate or atomicity that indicates the a priori probability that the specified proposition is TRUE in the absence of a committed belief mass.

To represent opinions graphically, SL adopts a two-dimensional equilateral triangle representation, presented in Figure *3-9*. A point inside this triangle represents an opinion. To clarify how to interpret the subjective logic triangle, we present on the left the reference axes and on the right the position of an example opinion.

Figure *3-9* Subjective Logic triangle of opinions




In the left-side triangle of Figure *3-9*, the belief, disbelief, and uncertainty axes run from the opposite side of the edge with the respective label, assuming the maximum value of 1 (one) in the edge with the label and zero in the opposite side. The probability axis is the bottom axis of the triangle, which is equivalent to the traditional probability axis because it represents opinions with zero uncertainty.

In the right-side triangle, we omit the axes and present an example opinion $\omega_x^A = (0.6, 0.3, 0.1, 0.5)$. The opinion represents 60% of belief mass, 30% of disbelief mass, and 10% of uncertainty. We also show the atomicity as a line dividing the triangle in the middle, indicating an equal 50% a priori probability of the proposition being TRUE or FALSE. Table *3-4* presents the equivalence between a subjective logic opinion and traditional probability theory.

The semantic of a subjective logic opinion can be better understood by means of an example. Imagine that the observer *Ricardo* wants to

quantify his opinion about a proposition $x$ related to his friend *Rodrigo*. The proposition $x$ states that "*Rodrigo arrives on time for his appointments*", meaning that *Rodrigo* is never late for any given appointment. In the absence of evidence, *Ricardo*'s opinion about proposition $x$ is of complete uncertainty, and is represented in subjective logic as: $\omega_x^{Ricardo} = (0, 0, 1, 0.5)$. This opinion means that *Ricardo* has no belief mass to support that the proposition $x$ is either TRUE or FALSE, and that from the complete uncertainty belief mass there is 50% atomicity, or a priori probability, that the proposition is TRUE or FALSE.

Table *3-4* Subjective logic opinions equivalence

| Subjective Logic | Equivalent To |
|---|---|
| belief = 1 | TRUE of binary logic |
| disbelief = 1 | FALSE of binary logic |
| belief + disbelief = 1 | traditional probability |
| belief + disbelief < 1 | degrees of uncertainty |
| belief + disbelief = 0 | total uncertainty |

Now let us imagine that *Ricardo*, after meeting with *Rodrigo* ten times, has experienced that in seven of these meetings, *Rodrigo* was on time. For the other three times, *Ricardo* himself was late, so he is uncertain about *Rodrigo*'s punctuality. Considering his experience, *Ricardo*'s opinion has changed, and his new opinion about proposition $x$ now is $\omega_x^{Ricardo} = (0.7, 0, 0.3, 0.7)$. This opinion means that *Ricardo* has 70% of belief mass about proposition $x$, and 30% of uncertainty. From the uncertain belief mass it is possible for *Ricardo* to assume that, if *Rodrigo* continues with his behavior, the a priori probability (atomicity) that he will be on time is now 70% for the uncommitted belief mass.

In SL, the probability expectation $E$ is calculate using the following formula $E_x^A = b + ua$. The intuition is that the expectation of an observer is the sum of the committed belief mass $b$, and the uncommitted belief $u$ mass multiplied by the a priori probability or atomicity $a$. Considering our previous example, *Ricardo*'s probability expectation about *Rodrigo*'s punctuality for the next appointment is $0.7 + 0.3 * 0.7$, which is equal to $0.91$. In summary, the atomicity represents how much of the uncertainty mass contributes to the probability expectation value that *Rodrigo*will be on time for the next appointment.

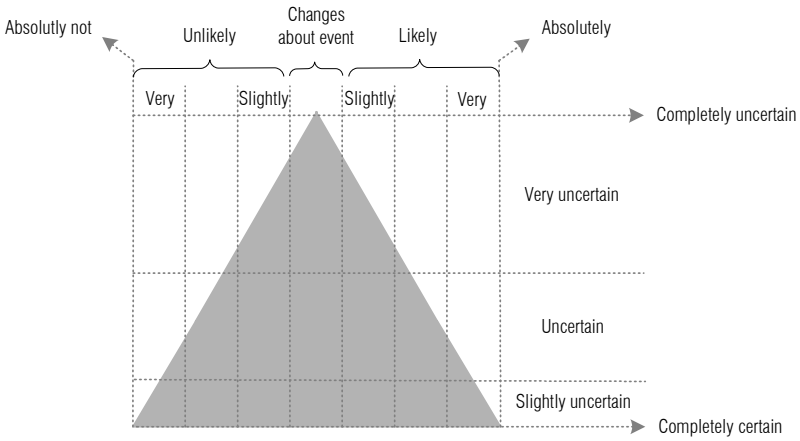
Subjective logic opinions can be also derived from the number of positive ($r$) and negative ($s$) previous experiences with an entity. The following formulas show how to calculate the belief ($b$), disbelief ($d$), and uncertainty ($u$) opinion parameters from previous observations. The weight ($W$) is usually instantiated to the value $2$ and represents the impact of new experiences in the opinion parameters [70].

$$b = \frac{r}{r+s+W} \quad d = \frac{s}{r+s+W} \quad u = \frac{W}{r+s+W} \quad W = 2$$

The advantage of subjective logic is not only the capability to express uncertainty, but also the many operators to compute over sets of opinions. Imagine that more than one person has different opinions about proposition $x$ - how can these opinions be combined for a final conclusion? The combination of the opinions could be done through the *consensus* operator, which provides a fair combination of opinions. SL also provides operators to perform subtraction and addition of opinions. Using a traditional probabilistic approach, the combination of a 100% probability with a 0% probability would result in a 50% probability, whereas when the *consensus* operator is used, the result would be complete uncertainty due to the conflicting opinions.

The SL logic literature suggests different ways of interpreting subjective logic opinions, using a set of concepts and mappings. Figure *3-10* shows the proposal division by Jøsang [71]. We believe that the mapping proposed by [71] is too fine grained for users to understand, and for this reason we propose a different and simplified mapping strategy in this thesis.

Figure *3-10* Fuzzy interpretation of SL triangle




We have mapped the subjective opinion triple (b, d, u) to an ordered set {very untrustworthy, untrustworthy, unknown, trustworthy, very trustworthy} whose elements model the judgment of user perspectives [100]. An opinion *op* whose belief is higher than its disbelief is considered trustworthy if it has an uncertainty of not lower than 1/3 and is very trustworthy otherwise. An opinion *op* whose belief is not higher than its disbelief is considered untrustworthy if it has an uncertainty of not lower than 1/3 and is very trustworthy otherwise (see Figure *3-11*). The *unknown* mapping represents opinions with even chances of equal belief and disbelief.

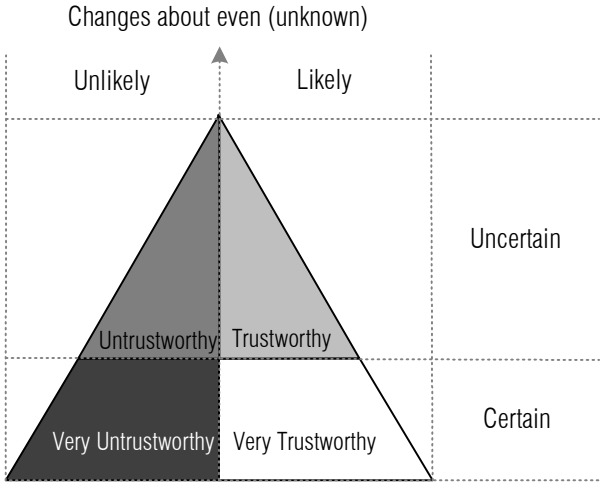



Figure *3-11* The function: [0, 1]3' VT, T, U, VU that maps an SL opinion onto the set of user judgments

***Evaluation of Trust Recommendations***

A trust mechanism for supporting the evaluation of trust recommendations is required because users and service providers may interact during context-aware service provisioning with entities that have unknown identities. If the identities are unknown, there are no trust values associated with this entities and no record of direct previous experiences exists. The approach adopted in this thesis for exchanging trust recommendations is inspired by the approach adopted by Almenarez et al. [3]. However, our approach is more complete because we take uncertainty into account and support the exchange of trust recommendations for the different trust aspects supported by our trust management model.

In the approach of Almenarez et al., the recommendation requests are broadcasted and the recommendation responses are combined taking into account only the responses of recommenders' with trustworthy identities. By using recommendations, indirect trust relationships can be established based on information received from other entities. Recommendation requests are only broadcasted when trustworthiness values are required for entities that are not known from direct experience or belief.

Recommendations can be received from trustworthy or untrustworthy recommenders. For this reason, it is important to support in our trust management model trust relationships related to the trust aspect of providing trust recommendations. Trust recommendations, despite being a trust aspect themselves, are also related to other specific trust aspects. For example, one entity might be trustworthy to give recommendations only about the privacy enforcement trust aspect about other entities. Recommendation trust degrees therefore state the amount of trust belief one trustor has in another trustee to provide recommendations about other entities with respect to a specific trust aspect.

In our trust management model, we merge the recommendations received from third parties using also the consensus operator from the Subjective Logic (SL), which has been proven to be a proper tool for this type of trust combinations [73]. The SL consensus operator is used to merge SL opinions in a "fair" way and, if conflicting opinions are received, the amount of uncertainty in the resulting trust degree is increased.

In contrast to the approach of Almenarez et al. [3] our proposal of combining recommendations using the SL consensus operator takes uncertainty into account. One major drawback of not considering uncertainty is a less accurate and less realistic trust result when conflicting recommendations are combined. Using the approach of Almenarez et al., when conflicting recommendations are received the result is an average of the belief probabilities. Using SL, when conflicting recommendations are received, there is an increase in the uncertainty.

Furthermore, Almenarez et al. [3] only consider the trustworthiness of the recommenders' identities and do not address trust values related to different trust aspects. In our approach, we subtract the trustworthiness of the identities from the trust recommendations, and we also support trustworthiness values for different trust aspects, including the trustworthiness of an entity to provide recommendations about specific trust aspects. For example, a trust recommendation received from a consumers' privacy protection organization will be influenced by the trustworthiness of the identity of the organization and by the trustworthiness value related to the organization's capability of providing recommendations about privacy enforcement trust.

In this thesis, we do not provide more complex algorithms for calculating trust from indirect knowledge. For more complex mechanisms, we refer to Toivonen et al. [115].

## 3.4 Mechanism for Selection of Context Providers

In this section, we present our mechanism for the trustworthiness evaluation of context providers. This evaluation is done by context consumers, which in our context-aware service platform are context-aware service providers. In our QoC model, trustworthiness is not addressed because it is not a quality attribute of the context information instance but a *degree of belief* from the point of view of the context consumer (e.g., the context-aware service provider) with respect to the context provider. The trust aspect of provisioning context information specifies the degree of belief of a context provider to provide context information about a context owner, and according to an advertised QoC level.

The bootstrapping of the trustworthiness values in our QoC model is done through pre-defined trustworthiness values or based on recommendations received from trusted third parties. Pre-defined trustworthiness values are usually defined based on the dispositional trust [88], that is, the likelihood of trusting other entities in the absence of concrete trust evidence. A simple and optimistic strategy would be to consider a context provider trustworthy by default, in case no recommendations are received or no evidence exists to believe the opposite. If multiple recommendations are received, they are combined using a "fair" combination, which is supported by the consensus operator from SL [69].

After the bootstrapping, the trustworthiness values evolve based on the feedback about the perception of users of the context-aware service regarding the reliability of its adaptation. When the users of the context-aware service notice wrong or inappropriate service adaptation, they can provide *negative* or positive feedback. Our feedback mechanism was inspired by the work of Huebscher et al. [61]. Positive feedback is mapped to a *trustworthy* opinion and negative feedback to an *untrustworthy* opinion. In case a positive feedback is received, the current trustworthiness value of the context provider for the specific context owner identity, context type, and QoC level is increased, and for the negative feedback, the trustworthiness value is decreased by the same amount.

The trustworthiness value decrease/increase is also computed by applying the consensus operator of the SL to the actual trustworthiness value of the context provider and to the feedback received. The behavior of the trust values range in the SL triangle of opinions is outlined in Figure *3-12*. Negative feedback only affects the trustworthiness of a context provider for a specific context type, context owner, and QoC level; in other words, it is possible for a context provider to be very trustworthy for one context owner and very untrustworthy for another.

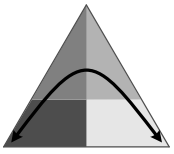


Figure *3-12* Behavior of trust values

If the context-aware service adaptation is not satisfactory, the service users have the possibility to indicate positive or negative experiences, and also indicate which faulty context-based adaptation behavior they are experiencing. Based on the specific feedback from the users, the service provider is able to detect which context provider is not fulfilling his promises regarding the quality of context and, therefore, the trustworthiness value of the context provider is decreased. The context-aware service provider may be able to detect, depending on the granularity of the feedback from the users, the exact context provider and context owner that is causing the faulty context-based adaptation. This detection is useful if context information from multiple context providers and context owners is being used by the service.

We do not support in our trustworthiness evaluation mappings of QoC levels and trustworthiness value combinations. For instance, if a context

provider is trustworthy to provide an *entitie's location* with $pm1$ *meter* precision, nothing can be said about the trustworthiness of the same context provider to provide the same *entities' location* with $pm1.01$ *meter* precision. We acknowledge that mappings between QoC levels and trustworthiness values are possible and depend on the context type; however, mappings of this type are outside the scope of this thesis.

In order to be able to collect relevant feedbacks, the context-aware service provider needs to be able to map the positive and negative feedbacks regarding the context-based service adaptation to the context provider that influences that behavior. The mapping of feedbacks allows a service provider to identify the reasons for a faulty context-aware service adaptation. For example, a negative feedback for a context-aware meeting service stating that one meeting attendee arrived to a meeting after the time predicted by the service may indicate that the person's location information provided was of low quality.

The positive and negative feedbacks capture the situation where the context provider provides context at a lower quality than advertised because he was dishonest, incompetent, and/or unreliable. The mapping from the user feedback (positive/negative) to the reason for lower-quality context information (dishonesty, incompetence, and/or unreliability) can only be evaluated if the granularity of the feedback includes enough detail about the faulty context-based adaptation. We only consider in our examples discrete positive/negative feedbacks without refining the faulty behaviors because they depend on the type of context-aware adaptation executed by the service.

The assumption we make is that, if the context-aware service is not adapting properly to the context, then the context provider is the one to blame. This might happen if the context information provided by the context provider capabilities is not as good as the capabilities being advertised. We support with our mechanism the situation where the context provider advertises a certain quality level and always provides context at a quality level lower than that advertised. In this case, the trustworthiness value of the context provider will never increase for the advertised QoC level because the trustworthiness value is increased according to our mechanism taking into account the QoC level that is associated with the context information instance. The assumption we make is that the context provider is not being dishonest because he still provides context according to the correct QoC specification.

We acknowledge that in some scenarios with a high number of context types and QoC levels it is not operationally feasible nor usable for users to manage the trustworthiness values for each combination. In scenarios where the number of combinations is too numerous, simplification strategies can be adopted; for instance, trustworthiness values for con-

text providers could be stored only considering the context types without distinguishing trustworthiness values for each possible QoC level.

## 3.5 Mechanism for Selection of Context-Aware Service Providers

In this section, we present our mechanism for the trustworthiness evaluation of the entities that collaborate in the provisioning of a context-aware service. This evaluation is done by service users, are also the context owners. In the description of this mechanism the assumption we make is that the context-aware service always uses the context information related to the service user in the context-based adaptation of the service provided (see Subsection 3.3.4).

Trustworthiness of identities

We define trust as a relationship between entities that are represented through digital identities. We therefore conclude that trust in a digital identity is influenced by the trust (regarding the trust aspect idp) in the identity provider that has provided that identity. The trust value associated with the provider of the trustee identity influences all the trust values associated with that identity. Our assumption is that it is not possible to trust the trust values associated with an entity who's identity is not trusted. This inter-relationship between trust in identities and trust in identity providers is synthesized by the following inference rule for indirect trust:

$$\frac{A \xrightarrow[v]{df;a} B \quad A \xrightarrow[v']{df;idp} C}{A \xrightarrow[v \otimes v']{if;idp} B} \quad role(C) = IP, \text{ C provides B's identity}$$

The above rule expresses the following: If $A$'s direct trust degree in $B$ regarding the trust aspect $a$ is $v$, and if the identity of $B$ is provided by identity provider $C$, and if $A$'s indirect trust in $C$ for aspect identity provisioning is $v'$, then $A$'s indirect trust in $B$ regarding aspect $a$ is the result of $v' \otimes v$, which represents the value $v$ subtracted by the value $v'$ (e.g. $v \otimes v' \leq v$).In the Subjective Logic (SL) domain we map $\otimes$ onto the fair combination *discount operator*.

The identity provider himself also needs to be identified through a digital identity. Therefore, we introduce a circular problem if the identity provider identifies himself with a self certified identity. For this reason, the previous rule does not apply for the identity provisioning trust aspect in case the identity provider provides his own identity. The trust value of an identity provider has to be determined by different means, for instance, through a pre-defined list of trustworthy or untrustworthy identity providers. The trustworthiness of a self signed identity can be

also directly mapped to the trustworthiness of the entity with respect to identity provisioning.

Once the service user has evaluated the trust relationships with all the entities that assume the context provider and service provider roles, the user deduces the combined trust value in the context provider (CP) role himself. The trust in the context provider and service provider roles has already been influenced by the trust the user has in the identity providers. This is a generalization step that allows the service user to evaluate his/her trust in the context provider role when the context-aware service retrieves context information from more than one context provider. The following rules express this generalization step.

$$\frac{A \xrightarrow[v]{if;a} C \quad [role(C)=CP]}{A \xrightarrow[v]{if;a} [CP,\{C\}]}$$

$$\frac{A \xrightarrow[v]{if;a} C \quad A \xrightarrow[v']{if;a} [CP,S] \quad [role(C)=CP]}{A \xrightarrow[v' \otimes v]{if;a} [CP, S \cup \{C\}]} \quad C \ni S$$

Here $a \neq idp$, because identity provisioning has already been in place. The rule on the top says that A's trust in the CP role can be initiated with the trust A has in members of the CP role. The rule on the bottom says that new members can contribute to A's trust in the CP role; so if A's trust in the role CP is v, and if A's trust in member C is v', then the new A's trust in the role is v v'. Here v v' expresses a "fair" combination of the two trust values as, for example, SL consensus operator.

The same generalization step for the context provider role could be applied for the service provider role in a similar way. For the service provider role, this generalization step would be required if more than one service provider were responsible for providing a context-aware service, for instance, in case of a more complex context-aware service composition. In our examples, we do not address these more complex cases; however, we acknowledge this possibility.

By using the consensus operator, we assume that all the entities that play the role of context provider have the same impact on the adaptation of the context-aware service and on the enforcement of the users' privacy. It is possible that two context providers contribute differently to the provisioning of the context-aware service and also that they have different impacts on the privacy of the context owners. We assume in this thesis that the impact is always the same, and we consider more advanced scenarios to be beyond our scope.

The final step of our mechanism consists of evaluating the service user's trust in the context-aware service as a whole. This evaluation depends on the trust the user has in both the CP and the SP roles regarding

the privacy enforcement and context provisioning aspects. The context provisioning aspect is only influenced by the members of the CP role. In this final step, we address two different user goal profiles derived from the trade-off between privacy enforcement and privacy adaptation. The first profile has higher priority in privacy enforcement and will accept less service adaptation. The second profile has higher priority in context-aware service adaptation even if privacy is not respected [9]. We name these two profiles privacy-focused and service-focused users.

The rule that expresses how to calculate A's (user) trust in a service provider B when context provider role is played by entities in S is formalized as follows:

$$\frac{A \xrightarrow[v]{if;pe} B \quad A \xrightarrow[v']{if;cip} [CP,S]}{A \xrightarrow[f(v,v')]{if;pe \times cip} B \times [CP,S]} \quad role(B) = SP$$

In this rule, the user combines his trust in the service provider role in the privacy enforcement aspect, and the trust he has in the context provider role in the context provisioning aspect. Function f expresses a particular way of aggregating trust, which depends on the two user profiles we address. In order to give an example of f, and for illustration purposes, we map TValues onto the ordered set $\{VT, T, U, VU\}$ (as described in Subsection 3.4.2) whose elements model judgment of user perspectives: very untrustworthy ($VU$), untrustworthy ($U$), trustworthy ($T$), and very trustworthy ($VT$).

Informally, Figure *3-13* shows the resulting trust in the context-aware service when the trust expectation in the service provider regarding the privacy enforcement aspect and the trust expectation in the context provider regarding the context information provisioning increase. The best case scenario for both user profiles is the one where the trust expectations for both the privacy enforcement and context information provisioning trust aspects are at least trustworthy. [8].

According to Figure *3-13* (a), for the privacy-focused profile the best cases are when privacy enforcement is at least trustworthy. The worst cases are when privacy enforcement is untrustworthy, because it is more likely that trustworthy context information about the user will be under a privacy risk. For the service-focused profile (Figure4 (b)), the best cases are when context provisioning is at least trustworthy, and it is even better when privacy is also enforced. The worst cases are when the context information is not trustworthy, which results in bad service adaptation.

[8] We assume that the trustworthiness value for the provisioning of context information is always associated to the same Quality of Context (QoC) level because it is outside the scope of this thesis to provide mappings of QoC levels and trustworthiness value combinations.

However, in this case, it is preferable to have privacy enforcement, if possible. We assume here that a context-aware service receiving untrustworthy context information is more likely to adapt wrongly to the current user situation. From this discussion we support the conclusion that for both user profiles, the best case is when trust in context information and privacy enforcement is high; however, depending on the profile, the worst-case scenario is not the same.

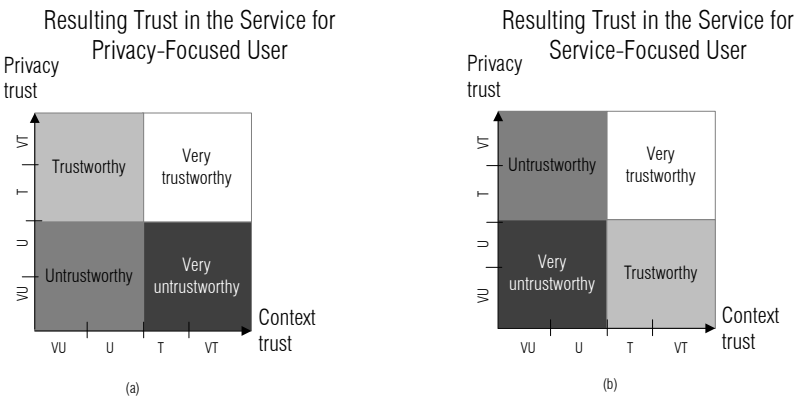



Figure *3-13* Resulting trust in the service according to a user profile that focuses on privacy (a) and on a user profile that focuses on service adaptation (b)

An example of the function $\int$ can be obtained by first applying $\pi$ to $v$ and $v'$, then applying one of the functions of Figure *3-13*, and then mapping back each user category onto a "representative" opinion of that category. For example a representative opinion of $VT$ can be the triple $(0.75; 0.01; 0.24)$, of $T$ it can be $(0.50; 0.01; 0.49)$, and so on. To the best of our knowledge, functions with the properties sketched in Figure *3-13* cannot be obtained by composing existing SL operators with $\pi$.

It is also possible that the trust values for privacy enforcement and context provisioning are unknown. If both values are unknown, we assume that the resulting trust in the service provided will be unknown as well. If only the trust value for the privacy enforcement trust aspect is known, then the resulting trust for the privacy-focused user is the value of this aspect and for the service-focused user the resulting trust is unknown. If only the trust value for the context provisioning aspect is known, then the resulting trust for the service-focused user is mapped to the value of this aspect and for the privacy-focused user the resulting trust is unknown.

If the value is unknown, another possibility would be to inform the privacy-focused user of the risk (s)he is taking by using the service. For instance, if the trust value for the context provisioning trust aspect is high, and the value for privacy enforcement is unknown, then the risk the privacy-focused user is taking is high. For the service-focused user, privacy is considered secondary; therefore, this risk approach does not apply.

## 3.6 Prototype Implementation

In this section, we describe a reference trust management architecture and a prototype implementation that demonstrate the technical feasibility of our trust management model. Our prototype implementation provides a graphical user interface to support users and service providers in the management of trust relationships and trust-based selection of context-aware service providers, context providers, context situation providers, and identity providers. The main objective of our prototype implementation is to demonstrate the decision support of the two mechanisms we propose for selection of context providers and context-aware service providers. The trustworthiness evaluation in our prototype is implemented using the Subjective Logic (SL) API [69] for trust calculations based on SL opinions. In the following subsections we describe our reference trust management architecture and the prototype that instantiate this architecture.

### 3.6.1 Trust Management Architecture

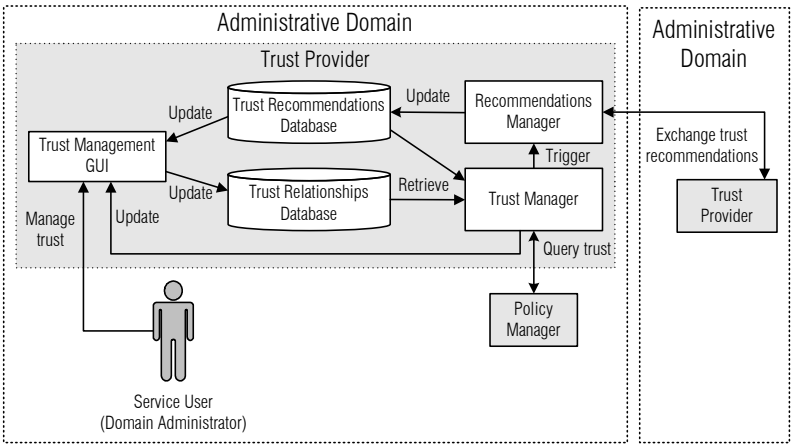
A context-aware service platform is typically a distributed system without a unique central point of control. In such a system, multiple administrative domains may exist. To illustrate this, consider a weather service that provides the local weather forecast for mobile phone users based on the latitude/longitude of the GSM cell they are located in. In this service example, the weather service provider, the mobile phone operator, and the user's personal device are examples of different administrative domains controlled by different administrative entities.

In this multi-administrative domain scenario, each administrative domain has a different domain administrator who is responsible for the specification of the domain security policies and trust relationships. Because the trust relationships are sensitive information, each administrative domain is interested in managing its own trust relationships with the entities in other administrative domains they collaborate with. In order to support the trust management tasks in a context-aware service platform, we propose the trust management architecture presented in Figure *3-14*.

Our trust management architecture supports distributed trust management considering that each administrative domain has its own *Trust Provider* component. We have divided the *Trust Provider* component into sub-components that are responsible for the different trust management functionalities we have identified in our trust management model and mechanisms. The *Trust Provider* sub-components are: the *Trust Management GUI*, the *Trust Recommendations Database*, the *Trust Relationships Database*, the *Trust Feedback Database*, the *Recommendations Manager*, and the *Trust Manager*.

The *Trust Management GUI* provides a graphical user interface to support the *Domain Administrator* in the specification of trust relationships, in the visualization of the trust recommendations received from other entities' administrative domains, and in the input of trust feedback about other entities. The *Domain Administrator* consults the *Trust Management GUI* to decide which service provider to use from a list of discovered service providers from a *Service Registry*.

Figure *3-14* Trust Management Architecture




The *Trust Recommendations* and *Trust Relationships* databases store direct and indirect trust relationships using trustworthiness values related to specific trust aspects as defined in our trust management model. Direct trust relationships are specified by the *Domain Administrator* using his own trust beliefs. Indirect trust relationships are trust relationships established based on trust recommendations received from other administrative domains by the *Recommendations Manager* sub-component. The *Recommendations Manager* executes the trust management mechanism for trustworthiness evaluation of trust recommendations.

The *Trust Feedback Database* stores positive and negative feedbacks related to trust relationships. This feedback can be used to increase/decrease trust values following a trust management strategy when the domain administrator observes entities that do not comply with the trust values in the trust relationships and trust recommendations databases. In one instance of our trust management architecture, the *Trust Feedback Database* is used to store feedback related to the QoC level advertised and provided by a context provider following the feedback mechanism we propose in our mechanism for trustworthiness evaluation of context providers.

The *Trust Manager* sub-component is responsible for coordinating the trust management tasks and for implementing the mechanisms for trust-based selection of context providers and context-aware services. The *Trust*

*Manager* sub-component uses as input the trust relationships specified in the trust databases, executes the trust management mechanisms, triggers the request for trust recommendations from other administrative domains, triggers updates of the *Trust Management GUI*, and provides an interface for other components in the administrative domain to query trust values. In Figure *3-14*, for example, a *Policy Manager* component queries trust values from the *Trust Manager* sub-component to support a policy management task.

### 3.6.2 Graphical User Interface

The user interface of our prototype implementation consists of a set of tabs, each one allowing the discovery of specific service provider types, namely: context providers, situation providers, identity providers, and (context-aware) service providers. For each service provider type a color coded trustworthiness value retrieved from the trust relationships database is shown. The colors we adopt are green, light green, gray, light red, and red to specify respectively a very trustworthy, a trustworthy, an uncertain, an untrustworthy, and a very untrustworthy trust degree.

A double click in the respective provider shows the trust belief screen allowing the trust relationship details to be modified. In the trust belief screen the trust value for the trustee identity is highlighted with a color code indicating the trustworthiness of the respective identity. Finally, the trust belief also optionally shows the corresponding mapping of the trust degree to the subjective logic triangle following the mapping proposed by us. The trust evaluation mechanisms and mappings implemented in our prototype are detailed in the description of our trust management model in Section 3.3.

The prototype we describe also supports integrated context-based management of trust and privacy policies in the tab *Context-based Policies*. The context-based support for policy management is described in Chapter 4 of this thesis.

#### *Trust-Based Selection of Context Providers*

Figure *3-15* shows the context providers tab and table with the discovered context providers in our simulated scenario. Context providers can be discovered considering the context type and respective context owner of interest. For example, it is possible to discover all available location providers for a specific user. In Figure *3-15* the discover result of all context providers available for the user identity *Ricardo Neisse* are shown in the table. For each context provider the description, context type, and trustworthiness value for the context provisioning trust aspect is presented.

When double clicking a specific context provider the trust belief screen is shown. In this screen it is possible to modify the trust relationships for

the respective context provider. Figure *3-16* shows the trust belief screen when double clicking the *Mobile phone GPS Location* context provider. In Figure *3-16* the editing of the trust aspect context information provisioning is shown. When editing relationships for the context provision trust aspect the associated QoC attributes are also specified. In this example the context provider is very trustworthy to provide GPS location with a precison of +-10 meters, a timestamp resolution in the order of seconds, and a refresh rate of 30 seconds. The refresh rate attribute is exclusively an attribute of the context information provider and is not part of the QoC attributes associated to context values.

Figure *3-15* Context providers discovery


Figure *3-16* Trust belief details for context information provisioning aspect

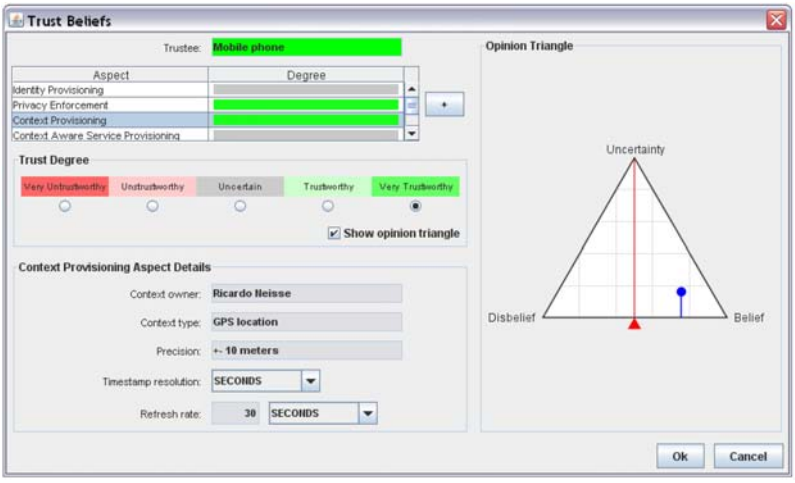



Figure *3-17* shows the trust belief screen for the privacy enforcement trust aspect. For this trust aspect additional attributes related to privacy enforcement are specified. In our prototype implementation it is possible to indicate whether the component under the user administration, the

presence of a policy enforcement point, and the presence of an enabled TPM chip.

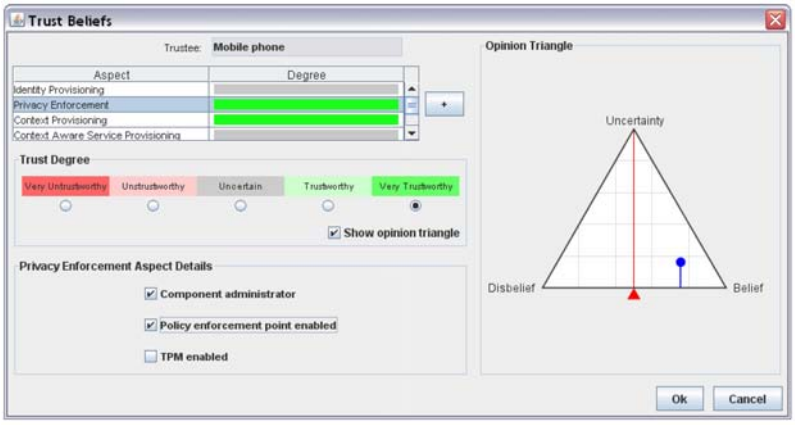



Figure *3-17* Trust belief details for privacy enforcement aspect

Our context providers tab also allows a selected context provider to be queried for context information. Figure *3-18* depicts the query result for the *Ambient Temperature* context provider. The query result shows the context value, QoC attributes, and a trustworthiness feedback option. After reviewing the context value a negative or positive feedback value can be provided, which triggers an increase/decrease of the trustworthiness value.

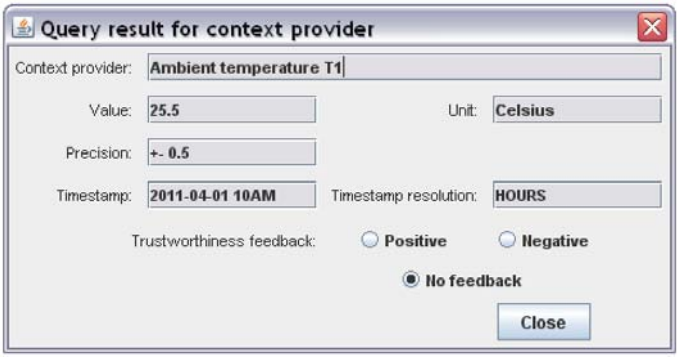



Figure *3-18* Context information query result

In addition to managing the trustworthiness of context providers we also support context situation providers. Situation providers are responsible for detecting situation events representing the moments a situation begins and ends to hold. Figure *3-19* shows the discovery interface for context situation providers. In the example discovery result all situation providers able to provide situation events for all situation types about the target entity *Ricardo Neisse* are shown.

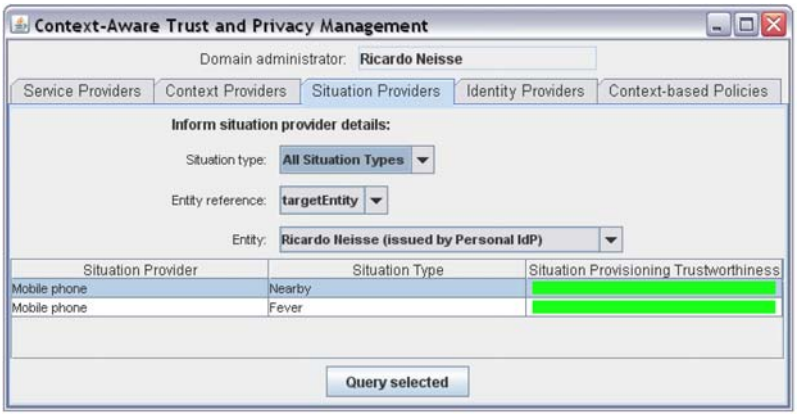



Figure *3-19* Situation providers discovery

The trustworthiness assessment of a situation provider is related to the signaling of situation events for a specific situation type and entity reference. Figure *3-20* shows the trust belief editor for the situation provisioning trust aspect. In this example the situation provider is considered trustworthy to report and detect situation events with a maximum delay of 10 seconds. These trust relationship is only valid for the *Nearby* situation type and the target entity *Ricardo Neisse*.

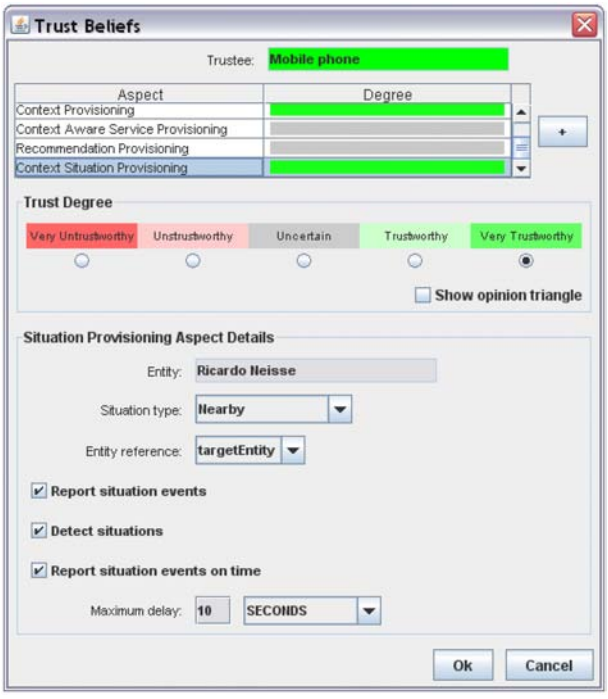



Figure *3-20* Trust belief details for situation provisioning

### *Trust-Based Selection of Identity Providers*

Figure *3-21* shows the identity providers tab and table showing the discovered identity providers in our simulated scenario. In this screen two identity providers are very trustworthy and one is trustworthy. When double clicking the *Personal Identity Provider* the trust belief screen in Figure *3-22* is shown. For the identity provisioning trust aspect the trust relationship restricts the trust relationship to a specific administrative domain. For example, the *Personal Identity Provider* is very trustworthy to verify identities only of entities from *Ricardo's Domain*.

Figure *3-21* Identity providers discovery

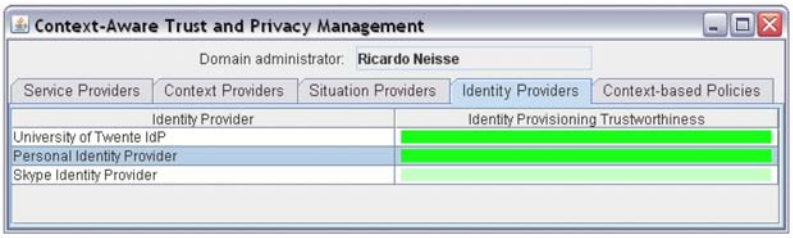
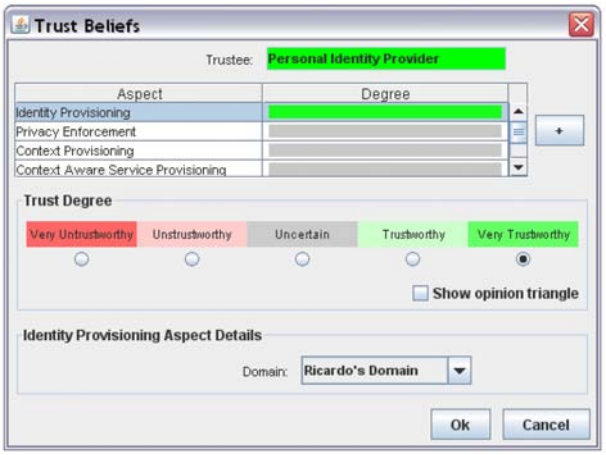



Figure *3-22* Trust belief details for identity provisioning

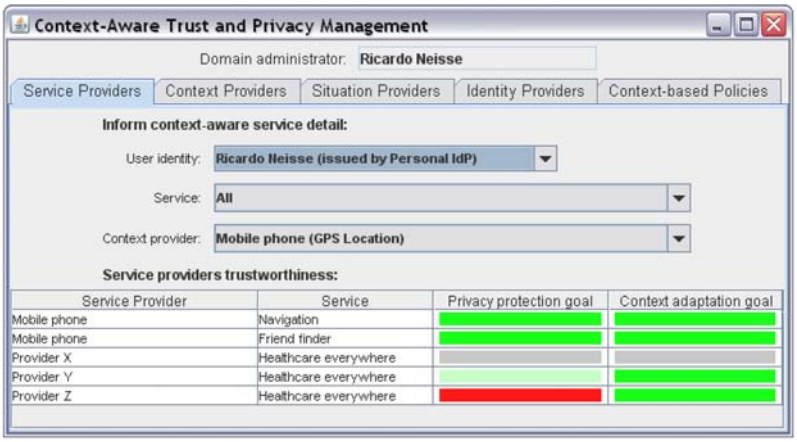



### *Trust-Based Selection of Context-Aware Service Providers*

Figure *3-23* shows the service providers tab and table with the discovered context-aware providers in our simulated scenario. Service providers can be discovered considering the service of interest. Additionally, it is necessary to select the service user identity from the list of available identities and the context provider from the list of available context providers. The table with the discovery results shows the list of service providers and respective trustworthiness values for the privacy protection and context adaptation user goals. These trustworthiness values are calculated using the mechanism implementation for selection of context-aware

service providers, taking as input the select user identity and context provider. The user is then able to selected based on his/her goal which service is more trustworthy considering his/her needs and goals.

Figure *3-23* Service Providers Discovery




The trustworthiness values in Figure *3-23* indicate that *Provider Z* and *Provider Y* are both very trustworthy considering the context adaptation goal. However, *Provider Z* is very untrustworthy to protect the user's privacy, and is therefore less desirable. When considering the trade off between privacy and context-based service adaptation for this specific context-aware service the best choice is *Provide Y* because it will adapt the service and also protect the user's privacy.

***Discussion and Lessons Learned***

The objective of our prototype implementation was to evaluate the technical feasibility of our trust management model and mechanisms to support users and service providers in selecting trustworthy entities to interact with. Service users are interested in the assessment of the overall trustworthiness of a context-aware service provider taking their primary goal into consideration. Service providers are interested in assessing the trustworthiness of context information providers to maximized their context-based adaptation capabilities. Our prototype implementation confirms that our QoC model and trust management mechanisms can be applied in practice and demonstrated the feasibility of these concepts.

With respect to the usability of our prototype implementation, we observe that the selection of context providers may be difficult if many fine grained QoC levels and trustworthiness values are specified. One important issue would be to allow context consumers to specify their QoC and trustworthiness requirements by means of ranges of trustworthiness values and QoC levels, and possibly by means of ranges of individual QoC attributes. For example, a context consumer could state a minimum and

maximum precision and a minimum trustworthiness value required for a context provider without specifying the absolute required values.

Based on our practical implementation we are unable to determined whether our trustworthiness evaluation mechanism is useful, or whether users understand the meaning and semantics of the trustworthiness values. This open question is addressed in the user survey described in Chapter 5 of this thesis.

## 3.7 Summary and Final Considerations

In the area of Quality of Context (QoC) modeling our contribution is a simplified and concise QoC model. Our QoC model is based on the existing literature and uses as a reference an existing ISO standard metrology vocabulary. Our QoC model clearly distinguishes the important quality attributes and clarifies the terminology of existing QoC models, providing a more accurate and simplified QoC vocabulary. This model directly benefits developers of context-aware service platforms because it improves the understanding of the QoC attributes. It is also part of our contribution to show how to apply our QoC model in a trust management mechanism to support context-aware service providers in selecting trustworthy context providers. Through our proof-of-concept prototype implementation we demonstrated the feasibility of our QoC model and show how it can be applied in practice.

We propose a new trust management model that supports the quantification of trust degrees for aspect-specific trust relationships that are relevant in our target context-aware service platform. Our model is extensible and considers trust aspects related to identity provisioning, privacy enforcement, context information provisioning, and context-aware service provisioning. We identified the interdependencies between these trust aspects and developed mechanisms based on a formalism for combining these trust aspects in order to evaluate the resulting trust users have in a context-aware service provider. We addressed two different resulting trust calculations considering the two different user goals we distinguish: privacy enforcement and context-based service adaptation. These goals are derived from the trade-off between privacy and context-based service adaptation related to the main goal of this thesis.

Based on our trust model and mechanisms specification we have designed and implemented a proof-of-concept prototype that demonstrates the technical feasibility of our contributions and show how our model and mechanisms can be used to assist users and service providers in their trust decisions and in the selection of trustworthy entities to interact with.

Our trust management model and mechanisms were designed to be applied in our target context-aware service platform. However, we foresee that our contributions could be easily applied to other service scenarios. Our contributions in the area of trust management are generic and extensible, and could be applied to other sets of trust relationships related to different trust aspects and trust management requirements. We have learned that in service oriented architectures trust relationships should focus on reliability issues of specific services, and may be combined in an overall trustworthiness assessment strategy that depends on the goals of the stakeholders considering the dependencies between the service providers.

Even though we do not support in our trust management model and mechanisms all possible scenarios, to the best of our knowledge we are the first to provide a systematic analysis of the trust issues in context-aware service platforms for different trust aspects. This analysis contributes to a better understanding of the problems and future research in this area.

Chapter 4

# Context-based Trust and Privacy Management

This chapter [1] proposes a new concept for supporting context-based management of authorizations and obligations, which we refer to as the *Context-Aware Management Domain* (CAMD). Our CAMD concept extends and integrates the context information model and *Context Handling Platform* developed by Dockhorn Costa [39] with the Ponder2 policy management framework [110, 116]. Our CAMD concept is motivated by the limitations of existing context-based policy management solutions that focus either on authorization or on trust management policies and do not support context-based obligation policies [75, 29, 27, 28].

Context-based obligations are important because privacy and trust management obligations in a context-aware service platform are triggered by changes in the context of the service users. This requirement is illustrated in the case studies of this chapter and on the user survey described in Chapter 5. With our CAMD concept we support integrated context-based management of QoC-aware authorizations, privacy obligations, and trust management obligations.

We show the *technical feasibility* of our CAMD concept by applying it in two case study prototype implementations: a *context-aware health service scenario* inspired by the AWARENESS tele-monitoring scenario [41] and an office-targeted context-aware service called *Colleague Radar* [92]. Our case studies confirm the *expressiveness* of our CAMD concept to support specification of context-based authorizations and obligations.

In the *Colleague Radar* case study we learn from a real context-aware service about context-based privacy requirements and show the design and implementation of an interface for user-centric context-based pri-

[1] Parts of this chapter have been published in papers [93], [94], and [97] which were co-authored by the author of this thesis.

vacy management. We also show how our CAMD concept implemented using Ponder2 can be integrated with the XACML policy model. The integration with XACML provides evidence of the *general applicability* of our CAMD concept in scenarios using other policy enforcement technologies than Ponder2.

This chapter is organized as follows. Section 4.1 presents our CAMD concept. Section 4.2 shows the context-aware health service case study. Section 4.3 describes the *Colleague Radar* case study. Section 4.4 concludes this chapter with a summary of the contributions and final considerations.

## 4.1 Context-Aware Management Domains

In this section we present an overview of our Context-Aware Management Domain (CAMD) concept, the information model, the component architecture, a description of the required steps to specify a CAMD, and a description of the information exchanged by the actors and components of our architecture.

### 4.1.1 Overview

The objective of our *Context-Aware Management Domain* (CAMD) concept is to support context-based management of trust and privacy preferences using authorizations and obligations. In order to fulfill this objective we integrate a general purpose policy management framework that supports authorizations and obligations with a context management platform. The result of this integration is our CAMD concept, which is essentially a context-aware extension of the *Management Domain* concept from the Ponder2 policy framework.

Context management platform

The context management platform selected by us to support the specification and operationalization of our CAMD concept is the *Context Handling Platform* (CHP) developed by Dockhorn Costa [39]. The CHP uses the context information model that is extended by us in Chapter 3 of this thesis for trust management purposes, including the concepts of entity, identity, context information, and context situation. In this chapter we use the architectural support provided by the CHP to realize these concepts that consists of Context Information Providers, Context Managers, and a Controller component. We chose Dockhorn Costa's model because of its expressiveness (see Subsection 2.1.1 and 2.1.3) and also because the CHP was available through an open source implementation that could be specialized and used by us in the first case study we introduce in this chapter.

Policy management framework

From the many general purpose policy-based management frameworks available (see Subsection 2.3.1 for a list) we chose the Ponder2

policy management framework [33, 32, 110] to support the specification and operationalization of our CAMD concept. We chose Ponder2 because it provides a generic policy information model that could be specialized by us, and because an open source implementation is available. For a detailed description of the Ponder2 framework we refer the reader to Subsection 2.3.1.

Figure *4-1* presents our target context-aware service platform and illustrates the deployment of policies in the service platform components[2]. In the picture, we divide the administrative domain into a *Management Layer* and an *Operational Layer* following the logical division of the Ponder2 framework. In the *Colleague Radar* case study the *Policy Administrator*, the *Service User*, and the *Context Owner* roles from Figure *4-1* are all assigned to the same entity, the person using the context-aware service.

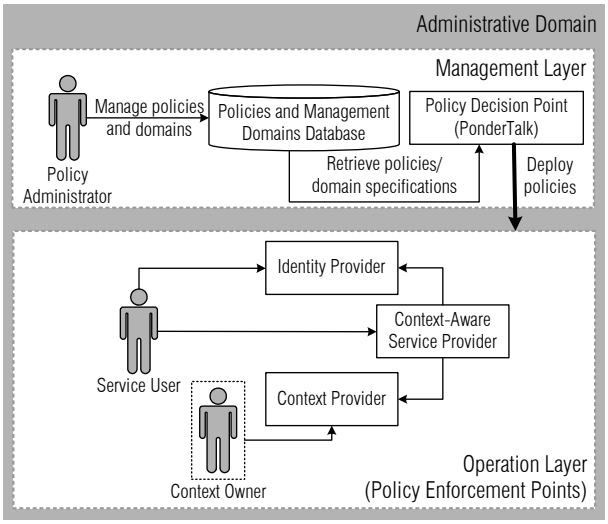



Figure *4-1* Policy Management in a Context-Aware Service Platform

The target enforcement point of the QoC-aware authorizations and privacy obligations we support with our CAMD concept are the *Context Providers* and *Context-Aware Service Providers*. The target enforcement points of the trust management obligations are the *Trust Provider* components deployed in each administrative domain as depicted by our trust management architecture in Section 3.6.

The Ponder2 policy information model specifies policy subjects and targets individually or in groups by means of management domains. However, even when using management domains, the set of entities that are members of a domain is static, and the system administrators have to as-

[2]We removed from the operation layer the description of the arrows representing the interactions between the components of the context-aware service platform. The meaning of the arrows is described in Subsection 2.1.4.

sign manually the entities to the management domains. Our concept of a Context-Aware Management Domain (CAMD) [93, 94] is a specialization of a *Management Domain* from Ponder2, which is explicitly associated with a *context situation* from the context information model we adopt.

A context situation captures a particular state of affairs specified through a set of entities and context conditions. When the context conditions evaluate to true, the situation begins to hold, and when the context conditions evaluate to false, the situation ceases to hold. Our CAMD concept is a sub-class of a management domain of Ponder2 that is populated with the entities that are part of a context situation. When changes in the *context situation* are detected by the *Context Handling Platform* (CHP) and events are generated to capture these changes in the context situations the CAMD membership, is also updated.

In order to illustrate the CAMD concept Figure *4-2* presents an example where the management domain "persons close to Ricardo" is mapped to the persons for which the location is at most ten meters away from Ricardo's location. In other words, a person becomes part of this domain if his/her location changes to a point equal to or less than ten meters away from Ricardo's location, and leaves the domain if his/her location changes to more than ten meters away from Ricardo's location. In order to monitor someone's location, the *Context Handling Platform* implementation provided to us assumes that the location of a person is the location of the person's mobile phone or personal GPS receiver.

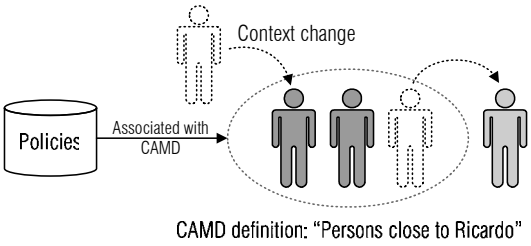



Figure *4-2* Context-Aware Management Domain (CAMD) membership changes

For the "persons close to Ricardo" CAMD, the system administrator can associate policies of different types, for example:

- Authorization: allow the entities that are in the domain (subject) access (action) to Ricardo's detailed contact information (target);
- Privacy obligation: when an entity leaves the domain "persons nearby Ricardo" (event), Ricardo's detailed contact information should be deleted (action);
- Trust management obligation: when an entity enters the domain "persons nearby Ricardo" (event), the trust degree for the aspect *people usually close to Ricardo* should be increased (action). Ricardo may decide to allow unrestricted access to his detailed contact information in the future to people with a high trust degree for this aspect even when they are not close by;

Authorization policies specify actions that subjects are allowed to perform on targets. Obligation policies consist of an event, a condition, and an action that is executed when the event is observed and the condition holds. For the exemple obligation policies described above (privacy and trust policies) the condition is not specified meaning that it is always *true* and the action is executed whenever the event is observed.

As illustrated in Figure *4-2*, a CAMD specification includes the specific context situation of interest and the set of policies that should be (de-)activated when the membership in this management domain changes. The (de-)activation of policies is managed automatically by the Ponder2 policy engine for traditional *Management Domains* and for CAMD in the same way. The difference is that for CAMDs, the membership is managed considering the changes in context situations.

In our examples, the entities in a context situation are always persons. However, we do not restrict entities to being persons. Context situations may be specified with respect to arbitrary entities, for instance, computing devices and buildings. This possibility is consistent with the CHP context information model.

### 4.1.2 CAMD Information Model

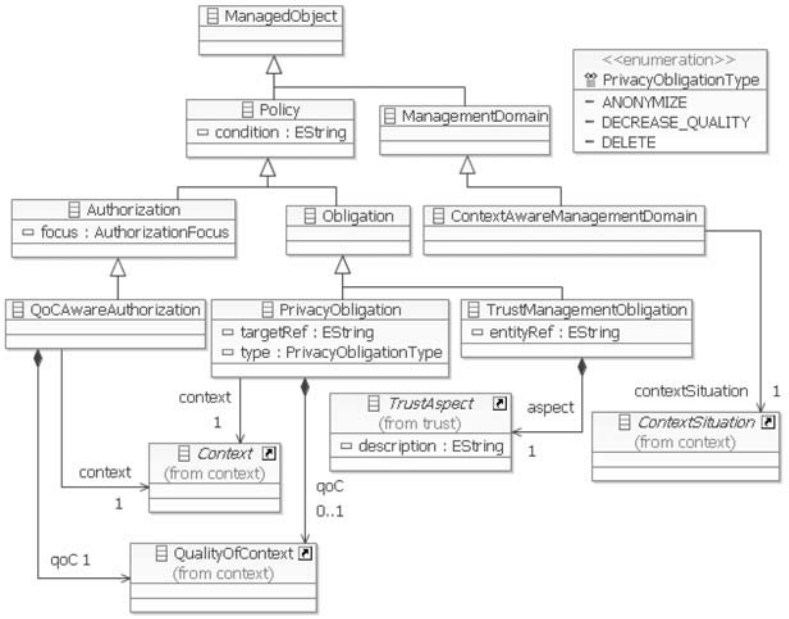
Figure *4-3* presents our CAMD information model. Our CAMD model defines a *CAMD* as a sub-class of the *Management Domain* class from the Ponder2 information model. The *CAMD* class is associated with a *Context Situation* class and, therefore, indirectly associated with the entities that are part of the situation. In order to allow the entities of our context model to be part of a management domain, we specify that a *Managed Object* is a possible role of an *Entity* class.

The association of policies with CAMD is done in the same way as specified by the Ponder2 information model, namely by means of targets and subjects. We specialized the Ponder2 information model to allow the definition of Quality of Context (QoC)-aware authorizations, privacy obligations, and trust management obligations, in addition to the already supported authorization and obligation policies by Ponder2 (Figure *4-3*).

*QoC-Aware Authorization* policies are a special type of authorization policies where the allowed action is access to a specific context type and a QoC level. A QoC-aware authorization policy specifies the maximum QoC level the context provider is authorized to provide access to. For example, *Ricardo* (target) may allow everybody (subject) access to his location (context type) but only at city level and not at street level (QoC level).

*Privacy Obligation* policies describe privacy protection actions that subjects must perform on targets under certain conditions. A privacy obligation policy could state, for instance, that a doctor must delete (action)

Figure *4-3* CAMD information model




all patient health data (targetRef) when the patient treatment is finished (event), or that the "identity provider" must anonymize (action) the patient's digital identity (targetRef) when a patient moves outside a hospital (event). We consider privacy obligations to delete context, to decrease context quality, and to anonymize context information instances.

*Trust Management Obligation* policies are used to manage the bootstrapping and increase or decrease of trust values regarding different trust aspects for a specific entity (entityRef). One trust management obligation policy could state that every time a doctor (entityRef) accepts to treat a patient (event), the trust in the reliability of the "treating patient" aspect of the specific doctor should be increased. The increase/decrease of trust degrees is managed using the feedback mechanism we already described in Chapter 3 of this thesis together with our aspect-specific trust management model, see Subsection 3.4.

### 4.1.3 Component Architecture

Figure *4-4* presents the architecture we define to support our CAMD concept. In our CAMD architecture, the system administrator specifies the CAMDs and the policies, and associates the policies with the CAMD. The CAMD manager component interacts with the Context Handling Platform by subscribing to context situations with Context Managers and then receives situation event notifications when the situations of interest begin and cease to hold. The CAMD manager is an specialization of the

Ponder2 Policy Decision Point (PDP) that includes event subscription and translation support to allow interoperability between the event formats of the CHP and Ponder2 framework. The *Policy Enforcement Points* (PEPs) in the CHP are the context manager and context provider components.

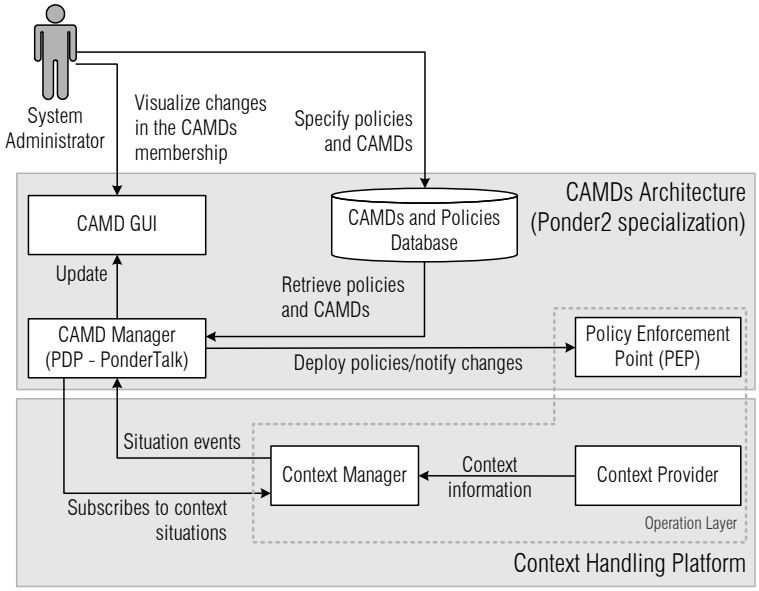



Figure *4-4* Context-Aware Management Domain (CAMD) architecture

The *CAMD Manager* subscribes to the *Context Manager* components from the CHP in order to be notified about the events related to the context situations of interest. We assume that the *System Administrator* is aware of the supported context situations and events in the CHP, and uses this context situation support as a tool to define the CAMD for the context-aware service scenario under his/her administration.

When the *CAMD Manager* receives the notifications from the CHP, it updates the working memory of the Ponder2 engine, updates the *CAMD GUI*, and notifies the PEPs. The *CAMD GUI* displays for the system administrator the management domains in a hierarchical tree, which includes the CAMD and static management domains. In the domain tree, the system administrator can also see the entities that are part of the domains and the policies that are specified in the system.

The system administrator in Figure *4-4* should specify and store the CAMD and policy specifications in the *CAMDs and Policies Database*. The *CAMD Manager* component reads the *CAMDs and Policies Database* and deploys the policies in the respective PEPs. In Figure *4-4* the PEPs are mapped to context managers and context information providers of the CHP. The PEPs for trust management obligation policies are the Trust Manager components deployed on each administrative domain.

In our CAMD architecture we do not include all the components of the Context Handling Platform. We do not use the *Controller* and *Action Resolver* components because our CAMD manager does not deploy ECA rules to trigger specific actions that could be delegated to the controller component. The CAMD manager receives all the situation events from the *Context Managers* and manages the changes in the CAMD membership internally.

### 4.1.4 CAMD Specification and Information Exchanged

The specification of a CAMD consists of specifying a management domain, the events that should be monitored by the Ponder2 engine, and the rules in the Ponder2 engine for adding and removing entities from the management domain when the event notifications indicating that a situation begins or ceases to hold are received. *Listing 4-1* illustrates the definition of the CAMD "Persons near Ricardo", the events that capture the change in the context situation that someone identified by an *identity* and *name* is near the entity *Ricardo*, and the rules that add and remove the entities to the CAMD when the event notifications are received. The listing also shows the creation of two simulated events when the situation *NearRicardo* is detected and begins to hold for the entity *Maarten*, and when the situation ceases to hold. The specification of the CAMD and the events should be done by the system administrator using the *Ponder Talk* language.

*Listing 4-1* Specification of the CAMD *NearRicardo* and the events and policies for managing the CAMD membership

```
// Create the CAMD for the persons near Ricardo
root at: "camds" put: domain create.
root/camds at: "Persons_near_Ricardo" put: domain create.

// Create the event description and parameters
root/event at: "EnterTrueNearRicardo" put: (event create: #("identity" "name")).
root/event at: "EnterFalseNearRicardo" put: (event create: #("identity" "name")).

root/camds/Persons_near_Ricardo at: "camdCreate" put: (obligation create).
root/camds/Persons_near_Ricardo/camdCreate
    event: root/event/EnterTrueNearRicardo;
   action: [ :identity :name |
      root/camds/Persons_near_Ricardo at: name put: identity].
root/camds/Persons_near_Ricardo/camdCreate active: true.

root/camds/Persons_near_Ricardo at: "camdDelete" put: (obligation create).
root/camds/Persons_near_Ricardo/camdDelete
    event: root/event/EnterFalseNearRicardo;
   action: [ :identity :name |
      root/camds/Persons_near_Ricardo remove: name].
root/camds/Persons_near_Ricardo/camdDelete active: true.

root/event/EnterTrueNearRicardo  create: #( "1" "Maarten" ).
root/event/EnterFalseNearRicardo create: #( "1" "Maarten" ).
```

After the CAMD is specified the system administrator can specify policies associated with the CAMD. *Listing 4-2* illustrates the definition of an authorization policy that authorizes all the entities that are members of the CAMD *root/CAMDs/PersonsNearRicardo* to execute the action *getIdentity* in the target entity *Ricardo*.

*Listing 4-2*
Specification of a policy that authorizes all the entities near the entity Ricardo to access Ricardo's identity information

```
// Create the Policy to allow nearby persons access to Ricardo's Identity
policy := authorization
  subject: root/camds/Persons_near_Ricardo
  action: "getIdentity"
  target: root/ricardo.
policy active: true.
root/camds/Persons_near_Ricardo at: "AuthIdentity" put: policy.
```

The following list summarizes the dynamics of a CAMD specification, including the entities that are involved, the interactions, and the information exchanged by the actors and components in the CAMD architecture described in Figure *4-4*:

- Using PonderTalk, the system administrator specifies the CAMDs, the structure of the events of interest from the CHP, the rules triggered by the events to manage the CAMD membership, and the policies referring to the CAMD, and stores these specifications in the CAMDs and Policies database;
- The CAMD Manager retrieves the CAMDs and policies specification from the CAMDs and policy database, subscribes to the CHP for the situations' events of interest, and deploys the policies in the PEPs;
- The CHP detects the context changes and notifies the CAMD Manager about the situation events;
- The CAMD Manager receives the event notifications, updates its working memory with respect to the changes in the domain's membership, notifies the PEPs, and updates the CAMD GUI according to these changes.

In our architecture, we do not address synchronization and communication problems, which may occur when context situation events are lost or when changes in context situations are generated during the specification and activation of a policy or CAMD. If the CAMD membership changes after a specific access has been granted we do not revoke the decision, only future requests for access will be denied.

## 4.2 Context-Aware Health Service Case Study

This section presents the first case study prototype implementation in which we support system administrators in the context-based management of QoC-aware authorizations, privacy obligations, and trust management obligation policies in a simulated context-aware health service scenario. The context-aware health service scenario is a simulated service and was not deployed in a real scenario. Our objective with this case study was to evaluate the technical feasibility of our CAMD concept and expressiveness to support the integrated specification of context-based authorizations and obligations.

### 4.2.1 Case Study Description

The objective of the context-aware health service is to improve the quality of life of epileptic patients. Patients with epilepsy have to be under constant vigilance in order to avoid dangerous situations because, during an epileptic seizure, these patients might experience convulsions and might hurt themselves. The context-aware health service improves the quality of life of epileptic patients by monitoring their vital signs and detecting or even predicting the occurrence of a seizure. When a seizure occurs or is likely to occur, the health service notifies the patient as well as nearby and available caregivers. Caregivers may be volunteers, usually family members or neighbors of the patient, who agree to help the patient when a seizure occurs. We implemented the following policies in our case study:

1. When an upcoming seizure is detected, caregivers who are nearby and have an available status are notified and allowed access to the precise patient's location (QoC-aware Authorization);
2. Only caregivers who accept to help the patient are allowed access to the patient's health data (Authorization);
3. Caregivers who accept to help a patient have their trust value increased for the availability trust aspect (trust management obligation);
4. When the seizure is over and the caregivers have completed their help to the patient, the access authorization to the health data should be removed and the caregivers are obliged to delete any data they have stored about the patient (privacy obligation).

### 4.2.2 Prototype Implementation

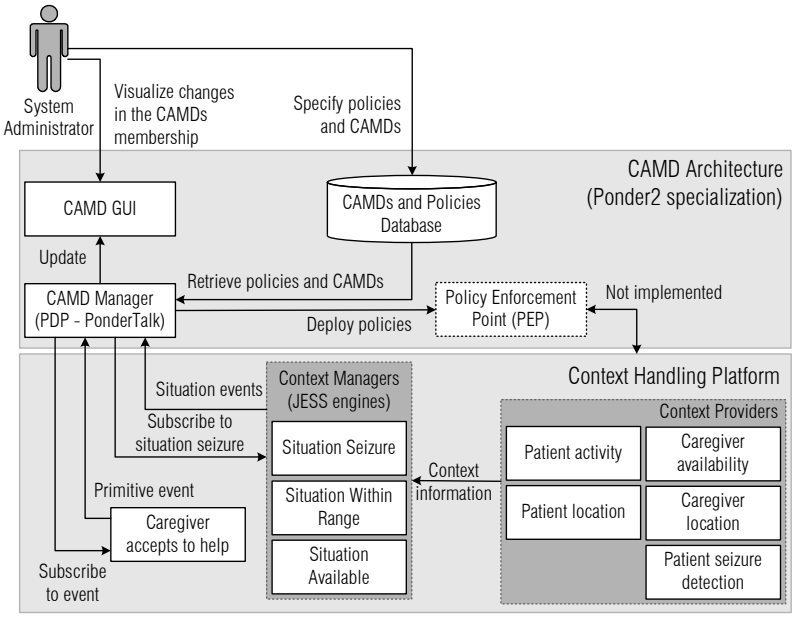
In this case study, the context information and context situations were generated using simulated context providers. Furthermore, the CHP does not implement the algorithms for detecting epileptic seizures. These algorithms are part of ongoing research, which was started in the AWARENESS research project [119] by the partner Roessingh Research and Development [109].

All the context information and context situation support for the context-aware health service was already implemented and provided to us by the authors of the *Context Handling Platform* (CHP). The CAMD and policy management support described in this section was implemented using our specialization of the Ponder2 framework. Figure *4-5* shows the components we implemented in this case study and the interactions between the components.

In Figure *4-5* the CHP supports context providers regarding *patient activity*, *patient location*, *patient seizure detection*, *caregiver location*, and *caregiver availability*. The CHP supports a primitive event provider *Caregiver accepts to*

*help*, which generates *primitive events* when a caregiver accepts to help a patient. The CHP also supports context managers that manage the situation seizure, the situation "within range", and the situation "available". The situation "within range" means that a caregiver is within a pre-defined distance of 1000 meters from the patient, and the situation "available" detects when the caregiver's availability has the value "on call". In the current implementation of the CHP, the *Context Providers* and *Context Managers* components are accessible through distributed objects in Java RMI and the Context Managers are implemented as rules in a JESS [20] rule engine.

Figure *4-5* Prototype screenshot




*Listing 4-3* presents the ECA rule we use in our case study to create the situation *Seizure* by the *Situation Seizure* context manager. When there is an epileptic alarm, a situation seizure is created and the patient and the caregivers who are within range of the patient and who are available are selected. This ECA rule, described in the ECA-DL language, has been translated manually by the authors of the CHP into JESS rules and was provided to us in their open source implementation. We included this ECA-DL rule to illustrate how the situation seizure in the health scenario case study is detected. For details on the ECA-DL language and all the JESS rules defined for the situations in this scenario we refer the reader to [39].

The CAMD Manager component in Figure *4-5* is implemented using the Policy Decision Point (PDP) from the Ponder2 framework. In our implementation we specialized the Ponder2 PDP to support event

*Listing 4-3* ECA rule for detecting the *Seizure* situation and selecting the patient and nearby and available caregivers

```
Scope (EpilepticPatient.*; patient) {
  Upon EpilepticAlarm (patient)
  Do Seizure(patient, Select (CareGiver.*; caregivers; (
      isCareGiverOf (caregivers, patient) and
      SituationWithinRange (patient, caregivers) and
      SituationCareGiverAvailable(caregivers))
    )
  )
}
```

subscription and translation for interoperability between the CHP and Ponder2 framework. The CAMD manager component subscribes to the context manager and converts the events received to events of the Ponder2 framework. Using PonderTalk, we implemented the policies for creating/deleting CAMDs and for deploying the respective policies of the health scenario.

The Ponder2 implementation we used does not provide support for remote policy enforcement points (PEP). Policies are centralized, evaluated, and enforced by the Ponder2 engine in the Java Virtual Machine that is executing the Ponder2 PDP. For this reason, we did not integrate the Ponder2 engine with the distributed context providers in the CHP. We could only experiment with the simulated situation events from the CHP and verify the deployment of the policies in our simulated environment. A newer version of Ponder2 released after we implemented our prototype support remote managed objects but it was not feasible for us to change our prototype after the new version of the Ponder2 framework was released.

Figure *4-6* shows the interface that simulates the context information developed by us for testing our CAMD implementation. This interface is integrated with the CHP and generates context information changes and primitive event notifications. The context information and primitive events are input for the Context Manager components that detect the specified context situations. When the CHP receives the context changes, and detects situations user JESS rules, our CAMD manager component is notified and possibly create/remove entities from the specified CAMDs.

Figure *4-7* presents our CAMD graphical user interface (GUI). The GUI displays the available domains in a tree structure on the left side and the details of each domain on the right side using tabs. For each domain, the tabs on the right side show the associated policies. We did not implement in our interface a visualization or authoring interface for the PonderTalk rules that manage the CAMD membership or policies. We also do not provide a management domain editor in our implementation. Previous versions of the Ponder framework [33] support a graphical domain editor and we expect that this will be available in future Ponder2 releases.

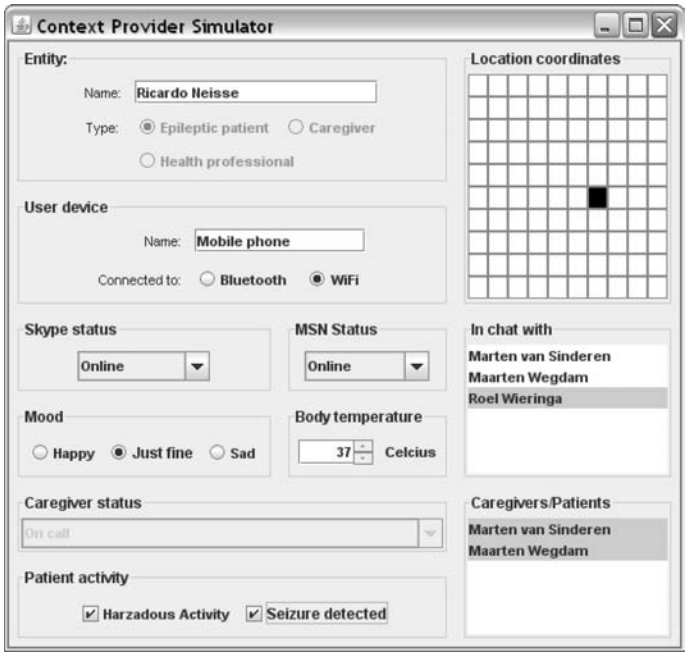



Figure *4-6* Context simulator interface integrated with Context Handling Platform (CHP)


Figure *4-7* Health scenario prototype graphical user interface

For illustration purposes, one of the PonderTalk policies defined in our prototype for creating the CAMD "Imminent Seizure" from Figure *4-7* is presented below (*Listing 4-4*). This PonderTalk policy creates CAMD sub-domains under the "Imminent Seizures" domain, upon the occurrence of an epileptic seizure alarm (EnterTrueSeizure). In this example, the hierarchy and the structure of the management domains follows a pre-defined model. The system administrator is free to design the most suitable CAMD structure to match the application scenario and the policy requirements. .

In our current prototype, the system administrator is responsible for the definition of the corresponding PonderTalk rules for managing the domain membership, and also for defining the appropriate CAMD structure considering the service scenario and the context situations of interest. The definition of these PonderTalk rules requires knowledge about the PonderTalk language [3]. The CAMD structure in our case study is composed of the patient having the seizure, the nearby caregivers, and the caregivers who accepted to help the patient.

*Listing 4-4* When EnterTrueSeizure event is received create the CAMD structure

```
camdCreate := obligation create.
camdCreate
event: EnterTrueSeizure;
condition: [1==1];
action: [
// Event parameters
:patient :startTime :nearbyAvailableCaregivers |
//
patientName := patient getName.
camdName := "Seizure_patient_" + patientName.
camd := createDomain value: root/Health_domain/Imminent_seizures  value: camdName.
//
domCaregiversAcceptedToHelp := createDomain value: camd
value: "Caregivers_that_accepted_to_help".
//
domNearbyAvailableCaregivers := createDomain value: camd
value: "Nearby_and_available_caregivers".
nearbyAvailableCaregivers do: [ :caregiver |
domNearbyAvailableCaregivers at: (caregiver getName) put: caregiver].
//
domStartTime := createDomain value: camd value: "Start_time".
domStartTime at: startTime put: startTime.
//
domPatient := createDomain value: camd value: "Patient".
domPatient at: patientName put: patient.

// CAMD policies go here
];
active: true.
root/policy at: "camdCreate" put: camdCreate.
```

The PonderTalk policy defined in *Listing 4-5* is triggered by the event generated by the context provider when the caregiver who is near a patient having a seizure accepts to help. This event is a primitive event and is not related to any specific context situation. A context situation could be defined if this were of interest to the service developer. This policy executes an action to increase the trust in the caregiver for the specific trust aspect of being reliable in accepting to help a patient having a seizure. Caregivers with high trust with respect to the availability trust aspect are considered more trustworthy to help in the future, and a low trust value for a caregiver might indicate that this caregiver should be replaced by a more trustworthy person.

The PonderTalk policy defined in *Listing 4-6* is triggered by the end of the situation "seizure". This policy removes the management domain from the hierarchy of domains, which also triggers an update to the graph-

[3] All the policies specified using PonderTalk for the *Health Service* case study are presented in the Appendix A.

ical user interface, and triggers privacy obligations. When the domain is removed the authorization policies are removed and all previous allowed authorizations to access to the patient's health data are revoked for the caregiver. This policy also executes privacy obligations to delete all patient data that has been accessed by the caregivers during the seizure.

*Listing 4-5* When CaregiverAcceptedToHelp event is received update CAMD structure and execute trust management obligation

```
camdUpdate := obligation create.
camdUpdate
event: CaregiverAcceptedToHelp;
condition: [1==1];
action: [

// Event parameters
:patient :caregiver |

patientName := patient getName.
camdName := "Seizure_patient_" + patientName.
root print: "Updating CAMD [" + camdName + "]".
camd := root/Health_domain/Imminent_seizures resolve: camdName.

root print: " - updating caregivers that accepted to help".
domCaregiversAcceptedToHelp := camd resolve: "Caregivers_that_accepted_to_help".

caregiverName := (caregiver getName).
domCaregiversAcceptedToHelp at: caregiverName put: caregiver.

root print: "Increasing trust in caregiver: " + caregiverName.
/root/trust/trustProvider increaseTrust: caregiver.

];
active: true.
root/policy at: "camdUpdate" put: camdUpdate.
```

*Listing 4-6* When EnterFalseSeizure is received delete CAMD structure revoking access to location and health data, and fulfill privacy obligations

```
// When CaregiverAcceptedToHelp is received delete CAMD structure
camdDelete := obligation create.
camdDelete
event: EnterFalseSeizure;
condition: [1==1];
action: [
// Event parameters
:patient |
//
patientName := patient getName.
camdName := "Seizure_patient_" + patientName.
root print: "Deleting CAMD [" + camdName + "]".
camd := root/Health_domain/Imminent_seizures resolve: camdName.
domCaregiversAcceptedToHelp := camd resolve: "Caregivers_that_accepted_to_help".
caregivers := domCaregiversAcceptedToHelp listObjects.
caregivers do: [ :value |
value deleteHealthData: (patient getName).
value deleteLocation: (patient getName).
].
camd removeAll.
root/Health_domain/Imminent_seizures remove: camdName.
];
active: true.
root/policy at: "camdDelete" put: camdDelete.
```

### 4.2.3 Discussion and Lessons Learned

Our objective with this case study was to evaluate the *technical feasibility* and *expressiveness* of our CAMD concept to support the specification of context-based authorizations and obligations. We used the context-

aware health service scenario to perform this evaluation. From the policy examples and CAMD implementation of the health service scenario we conclude that our CAMD concept is technically feasible and expressive to allow the specification of context-based QoC-aware authorizations, privacy obligations, and trust management obligations.

Our prototype was implemented for the simulated context-aware health service, and for this reason we can not be sure that our example policies are realistic and correspond to the requirements if this service would be deployed and used by real users. Our understanding of the policy management requirements was limited to the scenario we description we chose and the simulation we have performed using the implementation provided to us by the author of the Context Handling Platform [39]. As a result of this case study we identified potential examples of policies that could be defined for this service.

The policies in the case study were implemented in Ponder2, and deployed in the Ponder2 policy engine. We did not evaluate policy enforcement mechanisms for the different types of policies because n implementation of a policy enforcement component for Ponder2 external to the Java Virtual Machine running the Ponder2 PDP was not available. Because of this Ponder2 limitation, our prototype was limited to the specification and simulation of context-based policy dynamics without testing the enforcement of these policies in the context provider components. The enforcement of the policies in a distributed simulation impact the performance of the system and is out of the scope of this thesis.

## 4.3 Colleague Radar Case Study

This section presents the second case study prototype implementation. In this case study we provide support for users of a real context-aware office service, called the Colleague Radar service, to specify personalized context-based QoC-aware authorization policies. The context-aware office service was deployed at the Novay research institute in Enschede and was being used by their employees. Our objective with this case study is to evaluate the technical feasibility and to provide evidence that our CAMD concept can be used with other policy enforcement frameworks than Ponder2. The *Colleague Radar* service was already using the XACML policy framework for authorization enforcement. It was part of our objectives to learn about the service users' context-based privacy requirements and design an user interface for user-centric context-based privacy management.

### 4.3.1 Case-Study Description

The Colleague Radar is a service targeted at an office environment and was deployed for the employees of Novay in Enschede. The main objective of the Colleague Radar service is to help the Novay employees find each other when needed. Figure *4-8* shows the graphical user interface (GUI) of the Colleague Radar service.

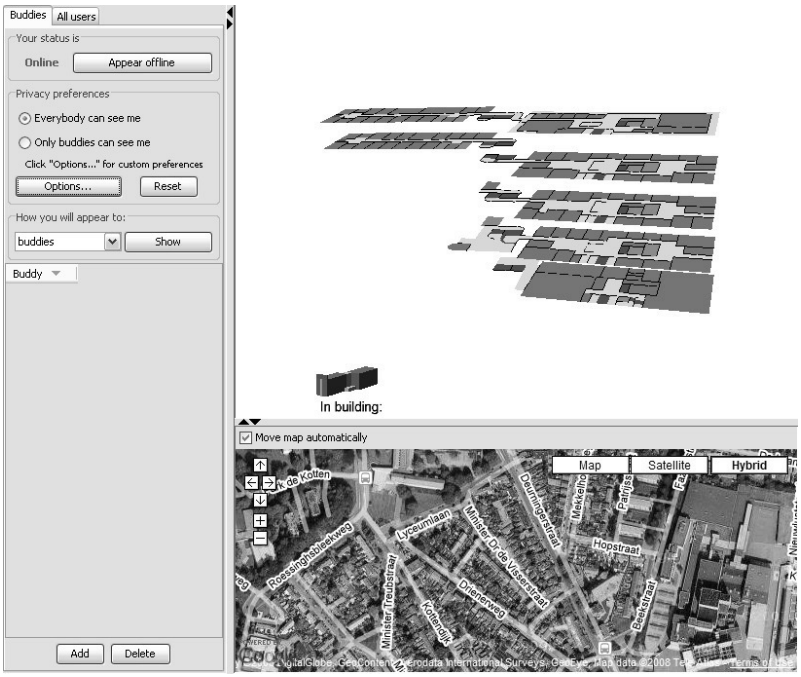



Figure *4-8* Colleague Radar graphical user interface

For each employee, the Colleague Radar GUI displays a list of buddies, which is a list of colleagues of interest using icons in different colors for each person. For each buddy, one can see his/her current location on a 3-D map when the buddy is inside the office building, and on a 2-D map (Google maps) when the buddy is outside the office building. The Colleague Radar service allows employees to also see other types of context, such as the schedule and appointments of the day and the instant communication status (e.g., online, out to lunch, busy). The additional context information is displayed by clicking on the corresponding icon of the buddy and might not always be available.

### 4.3.2 Design of Privacy Control Support

The privacy control mechanism we designed for the Colleague Radar service is divided into three different levels, which we named *main*, *basic*, and *advanced* levels. This three-level approach is recommended by user stud-

ies on privacy [120], according to the three different types of users: those who have almost no privacy concerns and should have coarse-grained control; those who are pragmatic and need a medium level of privacy control; and users who have a need for fine-grained control. Our main level allows coarse-grained granularity of privacy control and low complexity, while the advanced level presents the fine-grained granularity and relative high complexity for the users to adjust their privacy. User studies also show that users want a *switch off* option, to make themselves completely invisible at anytime.

The main and basic GUIs are presented in Figure *4-9*, on the left and right side, respectively. The main privacy screen is always visible for the users of the Colleague Radar service in the service client GUI (see also top left side of Figure *4-8*) and presents options to allow everybody access to the context information (default) or to allow only the buddies access to the context information. We have also included in the main view:

– An *invisible* option available in the GUI using the *Appear offline* button;
– An option to reset the privacy settings and change back to the default settings;
– A privacy preview option that shows to the user which context information a person in the selected group is authorized to see and allows users to visualize the effect of their privacy settings;
– An option to go to the basic privacy screen through the *Options...* button.


Figure *4-9* Colleague Radar main and basic privacy control GUI

The basic GUI (Figure *4-9*, right side) allows users to choose more fine-grained pre-defined policy templates than in the main screen. On the basic screen, users can select policy templates that are already context-aware to allow buddies or colleagues access to their context depending on the situation they or their colleagues or buddies are in. The templates we chose for the main, basic, and advanced levels are the result of a user survey [19] and represent the most common options mentioned by the users of the system on which they would like to allow access to their context information. From the basic view, users can also choose to access the advanced screen using the "Options..." button.

The advanced privacy screen (Figure *4-10*) is the most fine-grained control level for users, and is targeted at those users who are very concerned with their privacy. On this screen, users define personalized privacy preferences using a natural language template structure filling in the items of the template by selecting the appropriate context situations and QoC level. The personalized privacy preference template is composed of the following parameters: the context owners situation and time constraint (step 1), the context type and the allowed Quality of Context (QoC) level (step 2), the group and context situation of the colleague requesting access to the context information (step 3). Each policy template defines an authorization policy that permits access when all the parameters are matched in an authorization decision. Users are not allowed to specify negative authorizations, which simplifies understandability and conflict resolution because if one authorization policy evaluates to permit, access is granted and no conflict with a negative authorization will ever occur.

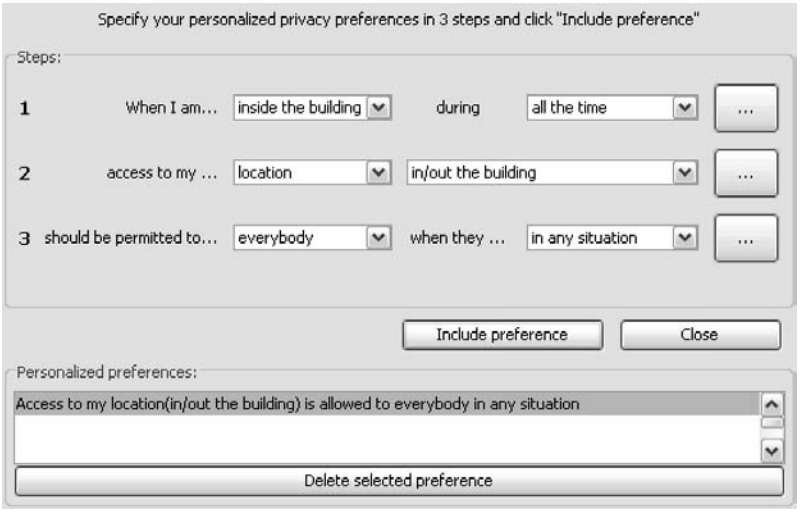
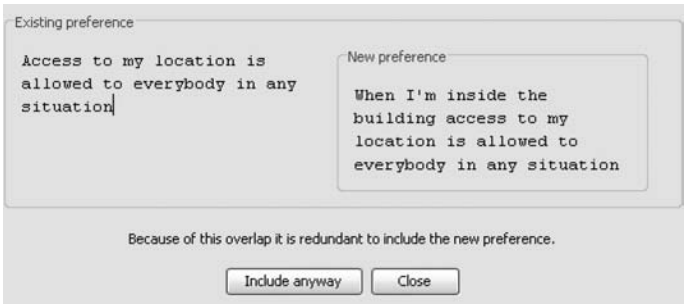



Figure *4-10* Colleague Radar advanced privacy control screen

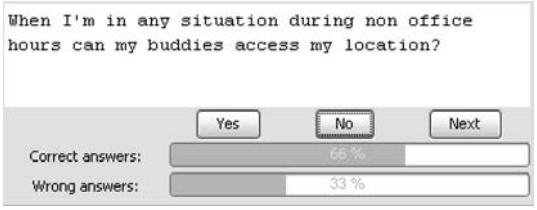
We also provide users with a basic preference overlap conflict resolution screen (see Figure *4-11*) that suggests to the user not to include a privacy preference when there is already a preference that overlaps with the preference that is being included. For instance, when a user has a preference stating that "all colleagues in any situation can see my location" and (s)he includes the preference "all colleagues inside the building can see my location", the second preference is a sub-set of the preference that is already defined. In this overlap case, the users is notified and (s)he may choose to include the preference anyway or to avoid the redundancy. We give this choice because users may want to specify redundant preferences in order to clarify their reasoning.

Figure *4-11* Policy overlap screen




In order to help users understand their privacy preferences, we also provide a Quiz question GUI (Figure *4-12*). The idea behind the Quiz questions is to allow users to verify if they understand their privacy preferences. The system randomly generates a combination of the parameters of the advanced privacy template and structures it in the form of a question. The question is then presented to the user, at least once a day, on the screen of the Colleague Radar application. The user can choose to answer or ignore the question, and he/she can also see the number of correct answers he/she has given. The answer is always "yes" or "no" and is related to the possibility of some group of users being able to access context information about the service user.

Figure *4-12* Quiz question screen




### 4.3.3 Prototype Implementation

The Colleague Radar service was implemented using the *Context Management Framework* [118], a framework developed by Novay, with native support for authorization policies in the *eXtensible Access Control Markup Language* (XACML) standard. The support for XACML in the CMF was implemented using Sun's XACML Implementation [42], and was limited to privacy policies defined by the system administrators without support for (1) user-based privacy preferences personalization and (2) context-based policies. Our objective in this case study is precisely to provide the users of the Colleague Radar support for these two aspects.

Our CAMD information model and architecture are a specialization of the Ponder2 policy framework and the *Context Handling Platform* (CHP), and for this reason we had to implement policy mapping functionalities

in order to use our CAMD concept in the Colleague Radar service. In addition to the mapping functionalities, we also designed a policy template mechanism, to allow user-centric authoring of CAMD specifications. This was necessary because no graphical policy editor was available that could be used by unskilled service users to define their privacy preferences.

Figure *4-13* presents the architecture of the Colleague Radar service, including the components of the CMF and of our CAMD architecture. The components from our CAMD architecture are integrated into the CMF framework and are presented in Figure *4-13* in the CAMDs framework box. In the Colleague Radar architecture, users of the service access the GUI to manage their profile information, to modify their privacy preferences, and to visualize context information about other users. Access requests for context information are redirected to the *Context User Agent*, which is instantiated in CMF for each user and is responsible for the access to all context information related to that user. The *Context User Agent* accesses the *Context Providers* and retrieves the context information that has been requested.

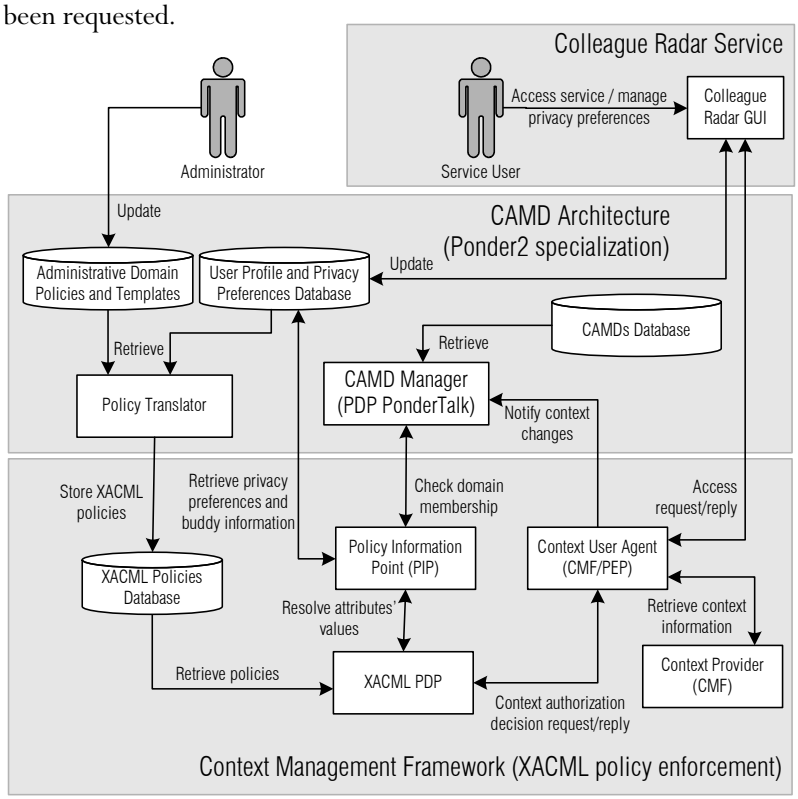



Figure *4-13* Architecture of the Colleague Radar context-aware service

The *Context User Agent* is the XACML Policy Enforcement Point (PEP) and makes context authorization decisions requests to the XACML Policy Decision Point (PDP). The XACML PDP retrieves policies that are

applicable to the authorization decision requests from the *XACML Policies Database* and evaluates the XACML policies. For each authorization decision request, the PDP replies to the PEP with an authorization decision, which can be either *Deny* or *Permit*.

XACML policies are composed of attributes and attribute values that are resolved through the *Policy Information Point* (PIP) component (Figure *4-13*). We encoded CAMD as attributes in the XACML policies and modified the PIP component to establish the CAMD attribute value (a.k.a. resolve) with our *CAMD Manager* component. The CAMD attribute indicates the CAMD membership of an entity for the specified CAMD attribute. For example, the attribute *camd:inbuilding* is resolved to *true* if the entity is currently a member of the respective CAMD *inbuilding*. The PIP component also resolves attributes from the XACML policies querying the *User Profile and Privacy Preferences Database* to verify, for instance, whether a user is a buddy of another user.

In Figure *4-13*, we also show the *Policy Translator* component, which is responsible for the translation of user-authored privacy preferences and domain policies to XACML specifications. Policies in XACML are written using the eXtensible Markup Language and are composed of rules that should be evaluated by the PDP. XACML policies are composed of lists of *Target*, *Subject*, *Resource*, *Action*, and an *Obligation* and can be grouped in a collection called *PolicySet*. The *Target* specifies to which authorization request the policy is applicable, the *Subject* specifies which entities are requesting the access, and the *Resource* specifies to which entities the access is being requested[4].

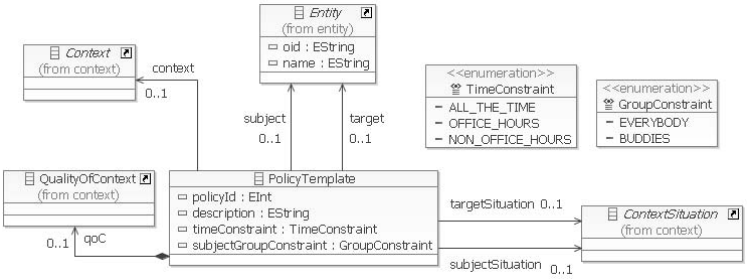
Actions in the CMF are mapped to access to context information. For instance, *Location* is an action that is mapped to the location of the employees. The *Obligation* in an XACML policy represents actions that must be performed by the PEPs when the authorization decision is returned. In CMF, the obligations in the policies are used to specify Quality of Context (QoC) degradations. The *Location* action could be combined with an obligation of *InOutBuilding*, which means that the exact location should not be revealed by CMF but only the relative location of the employee with respect to the office building.

Figure *4-14* shows the information model of the policy template we use to simplify the authoring of CAMD specifications. This policy template represents the specification of a QoC-aware authorization, and contains the subject and target of the authorization, the context type and QoC level, time and group constraints for the subject and target, and context situation for subject and target. An example policy for this template in natural language is: *Ricardo (target) allows all (subject) his buddies (subject group constraint) that are inside of the office building (subject situation) access to*

[4]For a complete description of XACML, we refer the reader to [45].

*his location in/outside of the building (context and QoC level) when Ricardo himself is also inside of the building (target situation) during office hours (time constraint)*. The policy template structure is precisely the structure of our advanced privacy control screen (see Figure *4-10*). Our *Policy Translation* component generates XACML policies from this policy template specification.

Figure *4-14* Model of the policy template for CAMD




*Listing 4-7* presents the respective XACML policy encoding for the policy that allows all colleagues of an employee that are in his/her buddies group access to the employee's location when the employee is inside the office building[5]. This template is encoded as one of the domain policies that are activated when the attribute *DomainPolicy_4* has any value or is present. This policy checks the group of the entity requesting access to the context information and the CAMD membership of the context owner.

*Listing 4-7* XACML policy for checking user group and CAMD membership

```
<Policy PolicyId="DomainPolicy_4" RuleCombiningAlgId="permit-overrides">
  <Description>DomainPolicy_4: Allow Buddies access if I'm in the office</Description>
  <Target>
    <Resources><Resource><ResourceMatch MatchId="function:string-equal">
      <AttributeValue DataType="string">true</AttributeValue>
      <ResourceAttributeDesignator
        AttributeId="preference:ColleagueRadar:DomainPolicy_4"
        MustBePresent="false" DataType="string" />
    </ResourceMatch></Resource></Resources>
    <Subjects><Subject><SubjectMatch MatchId="string-equal">
      <AttributeValue DataType="string">Buddies</AttributeValue>
      <SubjectAttributeDesignator AttributeId="subject:subject-group"
        DataType="string" MustBePresent="false"/>
    </SubjectMatch></Subject></Subjects>
  </Target>
  <Rule Effect="Permit" RuleId="DomainPolicy_4" >
    <Condition FunctionId="boolean-is-in">
      <AttributeValue DataType="boolean">true</AttributeValue>
      <ResourceAttributeDesignator AttributeId="camd:inbuilding" DataType="boolean"/>
    </Condition>
  </Rule>
</Policy>
```

All policies in CMF are configured with a *permit-overrides* policy combining algorithm because if any of the applicable policies in an authorization decision returns the permit decision as a result of its evaluation, then the resulting authorization decision should be evaluated to permit even if

[5] All the policies specified using XACML for the *Colleague Radar* case study are presented in the Appendix B.

other policies return deny as a result of their evaluation. We defined this combining approach because a default *deny* policy was specified matching all subjects and targets and the user specified policies were simply more specific policies allowing access. Therefore, if the user specified no policy all the access requests are denied as a result of the default policy. This approach is similar to a white-listing, all request are denied and specific requests that match a permit policy are allowed.

### 4.3.4 Discussion and Lessons Learned

Our case study shows that using CAMDs we are able to support user-centric management of context-based QoC-aware authorizations. The QoC-aware authorizations are personalized for each employee using the Colleague Radar service. We also show the *general applicability* of our CAMD concept through the integration with the XACML policy model.

We based the design of our context-based policies on the results of a user survey [19]. Considering the the results of this user survey we confirmed that there is a requirement for personalized context-based privacy preferences on different levels of granularity, which shows the relevance of our CAMD concept. The policies we specified are based on interviews and feedback of users and developers of the Colleague Radar service and provide a better understanding of the privacy requirements because it considers a real service scenario.

The Colleague Radar case study shows that context-based overlapping policies may be defined if users are empowered with personalized privacy preferences, and mechanisms and tools are needed to prevent redundant specifications. It is important that the users understand their privacy preferences and that they are able to verify their knowledge, preferably during the policy authoring task, to avoid the specification of redundant or conflicting policies. The verification of the users knowledge about their specifications can be done using the *Privacy Quiz* mechanism proposed by us.

## 4.4 Summary and Final Considerations

In this chapter, we introduced a new concept called Context-Aware Management Domain (CAMD). Our CAMD concept allows flexible and dynamic context-based policy management for context-aware service platforms. We confirmed the *technical feasibility* of our CAMD concept through two case studies: a context-aware health service and a context-aware office service. In our case studies we showed the that the CAMD concept is not limited to a specific policy implementation because we successfully applied it in two policy framework implementations: Ponder2 and XACML.

One major advantage of our CAMD concept is the integrated support for specification of context-based authorizations and obligations. We are not aware of any other context-based policy management mechanism that provides this support. In our case studies, we confirm the expressiveness of our CAMD concept through examples of QoC-aware authorizations, privacy obligations, and trust management obligations. Because this policies are associated to a CAMD, the subjects and targets are dynamically determined considering the detection of context situations.

In the *Colleague Radar* case study we propose policy templates to reduce complexity in user-centric policy authoring. Policy templates allow users to author policies in a straightforward and simple way. We have developed three levels of policy templates for QoC-aware authorization policies:

– A generic level for users unconcerned with privacy, with two options to allow or deny all the access;
– An intermediate level for users concerned with privacy with ten different policy templates already filled in;
– An advanced level for users very concerned with privacy, which allows fine-grained policy template specification.

We observed in the *Colleague Radar* case study that policy overlaps occur if users are allowed to specify fine-grain policies. Policy overlaps can not be detected in an efficient way at policy specification time when CAMDs are used. Overlaps can not be detected efficiently because the context situations associated to CAMDs are defined using arbitrary conditions over context information instances. As a result, CAMDs that depend on these situations might deploy overlapping policies with redundant or contradictory enforcement outcomes. For example, a policy may be specified to deny access to patient data when people are outside of the building and allow access to doctors on duty. It is not clear if a doctor on duty outside of the office building should be allowed access.

In our case study of the *Colleague Radar* service, we only allow positive authorizations; therefore, the only damage for users in case overlapping policies are specified is the confusion of having a redundant policy specification. For example, a policy allowing everybody access is redundant to a policy that allows buddies access because the access to everybody is already allowed. We were able to detect policy overlaps specified using our templates and implemented a mechanism to inform the users about the redundancy, letting them decide if they want to keep the redundant policy specification. In some cases redundant policy specifications might state more clearly for the users what is specified in their policies and in which context situations access is authorized.

We have also observed with our case studies that, depending on the complexity of the policies, it might be hard for users to know what policies are in place and which information is being accessed by the other entities.

To overcome this difficulty, we developed two tools that are promising on this respect: the *Privacy Quiz* and the *Privacy Preview* options. In a preliminary survey, users have positively evaluated these tools [19].

Chapter 5

# User Survey

The goal of this thesis is to support users and providers of a context-aware services in managing the trade-off between privacy (of context information) and context-based service adaptation. To achieve this goal our contributions are trust selection mechanisms and a context-based policy management concept called CAMD detailed respectively in Chapters 3 and 4 of this thesis. In this chapter, we describe the validation of these contributions through a web-based survey.

The focus of our survey is twofold; to evaluate the *usefulness*, *usability*, and *validity* aspects of our contributions, and to *learn* about the goals and choices of the survey participants. More specifically, the objectives of our survey were:

- **Usability**:
  - Evaluate whether the survey participants understand the concepts in our trust management model;
  - Evaluate if the survey participants understand the different roles in a context-aware service platform;
  - Evaluate if the survey participants understand the difference with respect to trust aspects when choosing the context providers, identity providers, and service providers;
  - Evaluate if the survey participants understand are able to provide their trust beliefs about these trust aspects for the different providers.
- **Usefulness**:
  - Evaluate if the survey participants would like to receive trust recommendations;
  - Evaluate if the survey participants agree that personalized context-based privacy preferences are useful.
- **Validity**:
  - Evaluate if the survey participants agree with the recommendations provided by our trust management mechanism for context information providers, identity providers, and service providers;

- **Learning:**
  - Learn about trust and privacy requirements in a context-aware service platform in order to better understand the trust and privacy goals of the survey participants, learn which trust and privacy issues are important for them when using a context-aware service, and get examples of context-based privacy preferences.

This chapter is further organized as follows. Section 5.1 describes our validation method and approach. Section 5.2 details the survey structure. Section 5.3 describes the profile of the survey participants. Section 5.4 shows the validation questions and the results of our trust management model and mechanism. Section 5.5 presents the evaluation questions and the results of our context-based management mechanism. Section 5.6 wraps up this chapter with a summary and final considerations.

## 5.1 Method and Approach

Validation method

The method we follow to evaluate the *usefulness*, *usability*, and *validity* of the results of this thesis is an general opinion user survey [44]. In our survey, we introduce a context-aware service scenario and ask questions about the choices and the trust beliefs of the survey participants with respect to the context providers, the identity providers, and the context-aware service providers. In the service scenario, the survey participants assume the role of context-aware service users.

Scope of the validation

Our survey consists of a set of open and closed questions designed to be self-administered through a web interface. We asked people to fill in our survey through an email message, which explains that our survey was about trust and privacy issues of a new service called Friend Radar. Participants of self-administered surveys have the tendency to stop participating in the middle of the survey (drop-off) if the survey is too long or complex [123]. In order to keep our survey simple, short, and to avoid long and complex explanations about terminology and concepts we:

- do not indicate explicitly in the email and in our survey that it was about a context-aware service;
- do not include QoC aspects because our contribution in this area are targeted at developers of context-aware services and not at end-users;
- do not include in our survey the validation of context-based management of trust relationships.

An evaluation of all our contributions would have increased considerably the complexity and size of our survey, due to the number of terms and concepts that would have to be introduced to our survey participants. We limit our survey to the validation of the concepts and trust management mechanisms we propose to support users of context-aware services

in selecting trustworthy context, identity, and service providers, and to the evaluation of our CAMD concept for context-based policy management through examples of authorization policies.

Validity and statistical significance

Because context-aware services are not yet widely available and it is not yet clear for us who the target population of context-aware services is, we applied our survey to the sample population that was readily available and convenient for us to reach. This type of sampling approach is called non-probability sampling, and does not allow us to make generalizations about the total population because the results would not be representative. Our survey was sent to colleagues, co-workers, research peers and collaborators. For these reasons, our survey results are only an indication of the usefulness, usability and validity of the contributions of this thesis and do not support statistically valid generalizations of our conclusions.

We sent an email asking people to fill in our survey to research collaborators, colleagues, and to the following universities, research projects, and research institutions mailing lists:

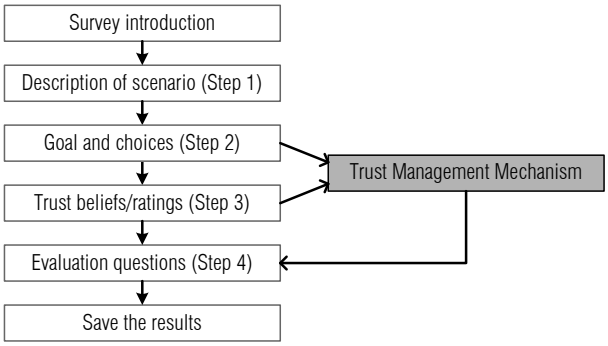
- Information Systems (IS) research group, Databases (DB) research group, Distributed and Embedded Security (DIES) research group, and Design and Analysis of Communication Systems (DACS) research group of the Computer Science Department of the University of Twente, Enschede, the Netherlands;
- Novay, Enschede, the Netherlands;
- AWARENESS research project;
- Federal University of Rio Grande do Sul (UFRGS), Porto Alegre, Brazil;
- Federal University of Vitória (UFES), Vitória, Brazil.

The complete text of our survey is described in appendix C. Figure *5-1* summarizes the structure we followed in the evaluation survey. Our survey steps were:

- Survey introduction: explains the goal and structure of our survey;
- Step 1: describes a context-aware service scenario for a service called Friend Radar;
- Step 2 - goal and choices: asks our survey participants what is important for them when using the Friend Radar service and their choices of context providers for location and activity information, identity provider, and service provider;
- Step 3 - trust beliefs: asks our survey participants to specify their trust beliefs for the providers from step 2 with respect to the trust aspects of context information provisioning, privacy enforcement, and identity provisioning. We refer to the trust beliefs in our survey as *ratings*;
- Step 4 - evaluation questions: shows to our survey participants the output of our trust management mechanism with respect to the

choices of providers taking into account our survey participants' goal from step 2 and ratings from step 3. We refer to the output of our trust mechanism as our *recommendation of providers*. In this step, we also asked evaluation questions related to our trust management model and CAMD concept;
– Save the results: expresses gratitude to our survey participants and save their results in a file.

Figure *5-1* Survey structure




***Survey Introduction***

In the introduction of our survey, we explain to the survey participants that our objective with this survey is to learn their choices and opinions with respect to the trust and privacy issues of a service called Friend Radar. We explain to the survey participants:
– *The functionality offered by the Friend Radar service*: the service runs on your mobile phone and allows you and your friends to visualize location and activity information about each other;
– *The completion time of the survey*: around 15 minutes;
– *Our privacy policy with respect to the survey data*: all data provided will be kept private and will be used only for research purposes;
– *The attitude we expect them to adopt when completing the survey*: the Friend Radar service is not available and we asked the survey participants to answer the survey questions to the best of their knowledge, imagining that they are a user of the Friend Radar service and of the technologies related to it.

We also asked the survey participants to fill in their email address, age, gender, whether their background is in computer science/engineering, their country of residence, and whether they are familiar with the following technologies: smartphone or PDA, GPS navigation, wireless networks, MS Outlook calendar, and Skype. The objective of these questions is to determine the profile of the survey participants.

***Step 1 - Description of the Friend Radar Service***

The Friend Radar service is an adaptation of the Colleague Radar service described in Chapter 4 of this thesis. We chose the Friend Radar service because it was an easy way to introduce to the survey participants the concepts and information providers involved in the provisioning of a context-aware service. Furthermore, this service scenario is adequate to illustrate the applicability of the models and mechanisms proposed by this thesis.

In step, 1 we [1]:

- explain to the survey participants the startup and login screens of the service and explain they can use digital identities from different providers; we state as examples *Google*, *Skype*, and the *university or company they work for*;
- explain to the survey participants that it is possible to see the location and activity information of their friends and other people, and that the availability of this information depends on their privacy preferences and vice versa: friends can see their location or activity depending on their privacy preferences settings;
- show to the survey participants the screens of the Friend Radar service to control their privacy preferences at three different granularity levels using groups, context situations, and personalized context situations;
- explain the privacy quiz and the privacy preview screens of the Friend Radar service.

***Step 2 - Choice of Providers***

In the second step of the survey, we ask the survey participants to answer five questions about the Friend Radar service. The questions are related to their goal when using the service and their choices of location, activity, identity, and service provider. Our objective with these questions is to determine which providers the survey participants would choose for the Friend Radar service. Our assumption was that different people would choose different providers based on their goal and trust beliefs. The questions and possibilities of choices in this step of the survey are:

- *Goal when using the service*: Your main goal when using the service is the protection of your privacy or the functionality of the service provided to you;
- *Location information provider*: you prefer that your location information is determined using your GSM cell location available through your mobile phone operator, your wireless base station location provided by your university/company, it does not matter (both sources are fine), or you do not understand the difference;

[1] For details, please consult appendix C.

- *Activity information provider*: you prefer that your activity information is retrieved from your calendar (e.g., MSOutlook) available through you university/company, your activity detected through sensors (e.g., accelerometers) in your mobile phone trough your mobile phone operator, it does not matter (both sources are fine), or you do not understand the difference;
- *Identity provider*: you prefer to be identified by your university/company, your Skype account, your Google account, anyone of the identities are fine, or you do not understand the difference;
- *Service provider*: you prefer to use the Friend Radar service provided by your university/company, by your mobile phone operator, by Google, anyone of the service providers are fine, or you do not understand the difference.

***Step 3 - Trust Beliefs***

In the third step, we ask the survey participants to rate their level of agreement or disagreement with a list of ten statements related to the Friend Radar service. The *ratings* correspond to the set of *trust degrees* we introduce in our trust management model described in Section 3.3 of this thesis, and the *statements* correspond to the different *trust aspects* for each of the choices of providers introduced in Step 2 of the survey. The trust aspects are the same we propose in our trust management model. We ask the survey participants to provide their ratings for the following roles and trust aspects:

- *Location information providers*: for the trust aspects of location information provisioning and privacy enforcement;
- *Activity information providers*: for the trust aspects of activity information provisioning and privacy enforcement;
- *Identity providers*: for the trust aspect of identity provisioning;
- *Service providers*: for the trust aspect of privacy enforcement.

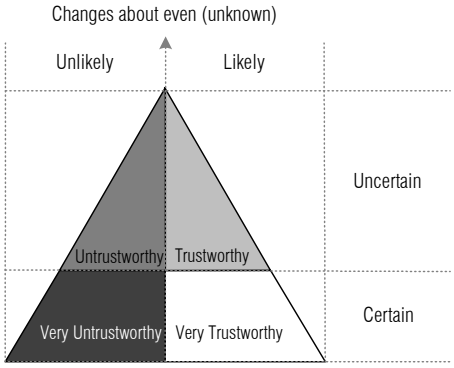
The ratings were labeled and explained to the survey participants accordingly to the semantics of the mapping of trust degrees to the Subjective Logic triangle of opinions also introduced in Section 3.3 of this thesis and repeated here in Figure 5-*2*.

We instructed the survey participants to interpret the ratings according to the following semantics:

- Strongly agree: I have reasons to believe in the statement and I am very sure. Equivalent to *Very Trustworthy* opinion;
- Agree: I have reasons to believe in the statement but I am not sure. Equivalent to *Trustworthy* opinion;
- Don't know: I do not have reasons to believe or disbelieve. Equivalent to *Unknown* opinion;
- Disagree: I have reasons to disbelieve in the statements but I am not sure. Equivalent to *Untrustworthy* opinion;

– Strongly disagree: I have reasons to disbelieve in the statements and I am very sure. Equivalent to *Very Untrustworthy* opinion;
– I do not understand this statement: the statement is not clear to me.

Figure *5-2* SL triangle partition




***Step 4 - Evaluation Questions***

In step 4, we displayed to the survey participants their choices for step 2 and evaluated if the survey participants' *choices of providers* were in agreement with the *output of our trust management mechanism*. The output of our trust management mechanism was generated using as input the survey participants' goals from Step 2 (privacy or service functionality) and the ratings from Step 3. We referred to the output of our trust mechanism in the survey as our *recommendation of providers*.

In this step of the survey, we were interested in evaluating the validity of our trust management mechanism to support users of context-aware services in selecting trustworthy service providers according to their goals. The output of our trust management mechanism is a list of trustworthiness values for context-aware service providers given as input the trustworthiness values for the identity provider, the context provider, and the service provider for the trust aspects of identity provisioning, context information provisioning, and privacy enforcement. For more details, refer to Subsection 3.6.1 of this thesis.

Figure *5-4* presents a summary of the informal reasoning behind our trust management mechanism that was implemented in our survey. This reasoning represents precisely the resulting trust calculation proposed in Chapter 3 of this thesis in Figure *3-13*. This figure is repeated here in Figure *5-3*. The objective of the recommendation is always to maximize the trustworthiness of the selected service providers.

The reasoning of our trust management mechanism depends on the selected goal of the user: *privacy* meaning a Privacy-Focused User, or the *functionality of the service* meaning a (Service-Focused User). The recommendation based on our trust management mechanism is the following:

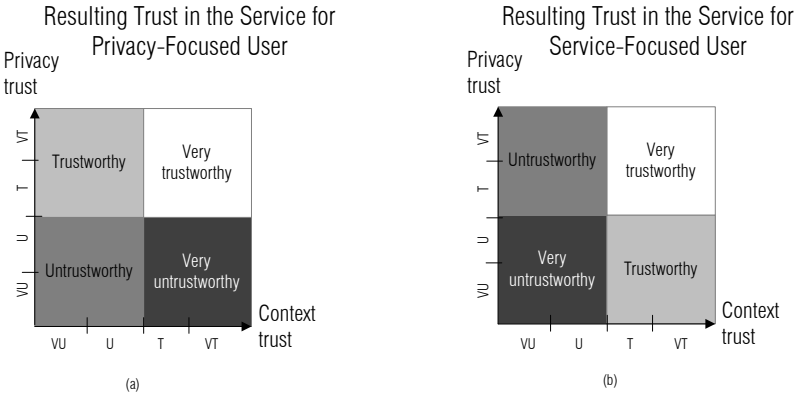



Figure *5-3* Resulting trust in the service according to a user profile that focuses on privacy (a) and on a user profile that focuses on service adaptation (b)

- If the goal of the survey participant is privacy, and the combination of the trust degrees (ratings) for the privacy enforcement trust aspect is unknown, trustworthy, or very trustworthy, the trust management mechanism recommends the MORE trustworthy identity and context providers together with the more trustworthy service providers for the privacy enforcement trust aspect. This strategy maximizes the trustworthiness of context and identities considering that privacy is protected;
- If the goal of the survey participant is privacy, and the combination of the trust degrees (ratings) for the privacy enforcement trust aspect is untrustworthy, or very untrustworthy, the trust management mechanism recommends the LESS trustworthy identity and context providers together with the more trustworthy (or less untrustworthy) service providers for the privacy enforcement trust aspect. This strategy minimizes the trustworthiness of context and identities considering that privacy is NOT protected;
- If the goal of the survey participant is the functionality of the service the trust management mechanism recommends the MORE trustworthy identity and context providers together with the more trustworthy service provider for the privacy enforcement trust aspect. This strategy maximizes the trustworthiness of context and maximizes the privacy protection.

We applied our mechanism to cover all the possible choices of providers according to the ratings provided by the survey participants in step 3, taking into account the survey participants' goal from step 2. This allowed us to compare the output of our trust management mechanism to the survey participants' choices for all the possible choices of activity information providers, context providers, identity providers, and service providers. Furthermore, it allowed us to assess individually which of their choices of providers is in agreement with the recommendation of our trust management mechanism.

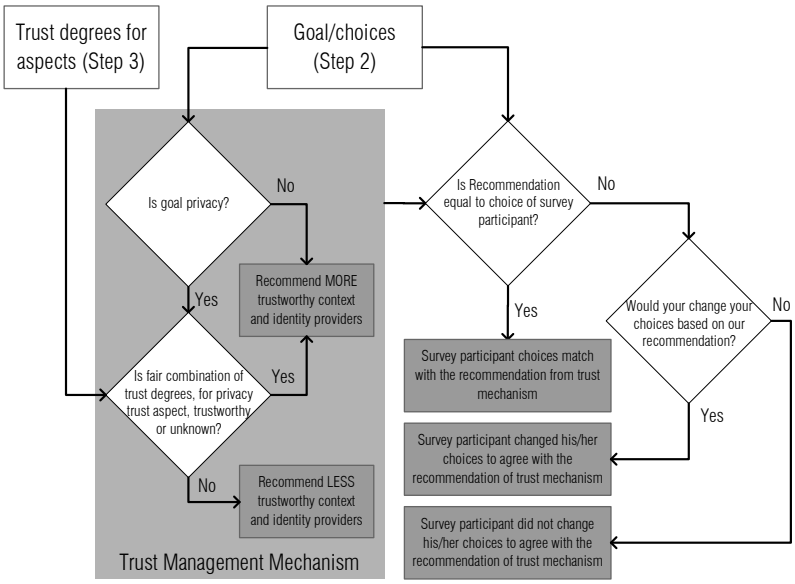



Figure *5-4* Reasoning of trust management mechanism in user survey

After displaying the output of our trust management mechanism, we explained to the survey participants which of their choices of providers were in accordance with the output of our trust mechanism and we explained why our trust management mechanism made the recommendation. For the choices of the survey participants with respect to the providers that were not in accordance with the output of our trust mechanism, we gave the participant the option to change his/her mind and agree with our recommendation. Based on this reasoning we could cluster the survey participants into three groups:

1. *The choices of the survey participant were in agreement with the recommendation of our trust management mechanism*;
2. *The choices of the survey participant were not in agreement with the recommendation of our trust management mechanism and he/she agreed to change his/her choices to agree with the recommendation after reading the explanation of the recommendation*;
3. *The choices of the survey participant were not in agreement with our trust mechanism and the survey participant did not accept to change his/her choices*.

In step 4, we provide the survey participants the opportunity to explain in open-ended questions why they decided to accept or not accept the output of our trust management mechanism. Furthermore, we also asked them 20 questions related to the evaluation of the usefulness and usability of the trust management mechanism and the context-based privacy mechanism described in the Friend Radar service in Step 1. In the evaluation of the context-based privacy mechanism the survey participants were only able to provide their opinion about what they understood

of the context-based privacy mechanism based on a graphical representation and a short description of the mechanism functionality. This part of the survey was not interactive and the participants were not able to use or experience the privacy mechanisms.

***Saving The Results***
After completing step 4, the survey participants saw a message of gratitude for their collaboration and their answers were saved for analysis.

## 5.2 Analysis of Survey Results

In this section, we present an analysis of our survey results. We do not include all the survey data we collected. The complete set of data we collected and analyzed is presented in Appendix B. The following list summarizes the profiles of the 60 survey participants:

- 145 people accessed the survey introduction, 113 step 1, 89 step 2, 85 step 3, 84 step 4, and 60 the final step, which is a 53% drop-off rate;
- The average time needed to complete the survey was 22 minutes;
- 38 of the survey participants identified themselves through their email address (64 %);
- 48 were males (80 %);
- 82% were in the age range of 20 to 40 years old, one person was less than 20 years old, and the remainder were more than 40 years old;
- 95% were from a computer science/engineering background;
- 75% resided in the Netherlands;
- 97% answered that they understand how the Friend Radar service works.

### 5.2.1 Trust Management Model and Mechanism

In this subsection, we present the evaluation questions and results of our survey with respect to our trust management model and mechanism for the recommendation of trustworthy providers.

***Choices of Goals and Providers***
Our evaluation questions with respect to the choices of the survey participants in Step 2 were:

- For how many people are the goals of privacy or the functionality of the service more important?
- How many people choose, do not care about, or do not understand the roles of location information provider, activity information provider, digital identity provider, and service provider?

– Do people find it difficult to choose the providers?
– Which other goals, in addition to privacy and functionality of the service, do people have in mind?

Our survey results show that 75% of the people think that privacy is more important, while the rest of the people (25 %) think that the functionality of the service is more important. About 50% of the people do not care about the provider of their location information, 25% do not care about the activity information provider, and about 20% of the people do not care about the identity provider and service provider they are going to use. A very small percentage of the people (less than 5%) did not understand the difference between the location, identity, and service provider roles. Our conclusion from these numbers is that almost all the survey participants could understand the roles in a context-aware service platform.

From Figure *5-5* we conclude that for the majority of survey participants (65%), it was not difficult to choose the location information providers, activity information providers, identity providers, and service providers.

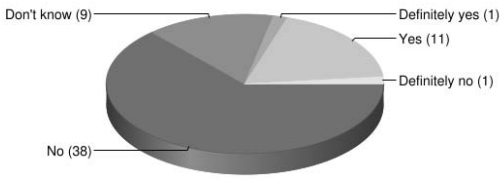



Figure *5-5* Pie chart showing how many of the survey participants found it difficult to choose the providers

The survey results show that a great majority of the survey participants (84%) think that goals other than privacy or functionality of the service are important. Our user survey included one open-ended question asking the survey participants to specify which other goals they believe are important. From the answers to these questions, we confirmed that our trust model is not complete with respect to the goals of context-aware service users and could be extended to cover more goals. The answers for this question were:

– Cost of the service free or provided at low costs. This goal was indicated by 10 survey participants;
– Battery usage;
– User friendliness and ease of use;
– Usefulness and added value in terms of time saving;
– Increase of fun;
– Accuracy, availability, reliability, scalability, quality of service;
– Possibility to get rid of the service. We believe the survey participant values the possibility of stop using the service;

- Accessibility, coverage of the area, which friends are visible (i.e., what do my friends use), not limiting the Friend Radar to a mobile phone and providing it also on the Web;
- Integration with other services, e.g., social networks, twitter, microblogging; leaving a message at a certain location, for example, if a buddy hits that spot, or is close by, a message pops up "You must see this museum, it is awesome"; updating status of the user on the web;
- Context-aware recommendation services: would function both as a value aggregator, if able to provide useful recommendations to users, and as an additional revenue generator to the providers (should be done carefully in order to avoid overwhelming the users with useless marketing). We believe an example is a restaurant recommendation service that recommends restaurants based on the user profile, location, and time of the day;
- Compatibility with all phone models;
- Safety, e.g., not revealing that my house is empty;
- Control, plausible denial, and embarrassment, e.g., don't want friends to know I don't exercise.

***Ratings or Trust Beliefs***

In step 3 of our survey, we asked the survey participants to state their trust beliefs, which we referred to in the survey as *ratings*. Our evaluation questions with respect to the trust beliefs of the survey participants in step 2 were:

- How many people understand the trust aspects of location information provisioning, activity information provisioning, identity provisioning, and privacy preferences enforcement?
- How many people find it difficult to provide their trust beliefs about the providers?
- How many people would like to receive trust recommendations about the providers?

From our survey results we conclude that the great majority of the users (more than 95%) understood the trust aspects and were able to provide ratings about them. Only 11% of the survey participants did not understand the ratings related to the provisioning of identities and less than 5% did not understand one of the other trust ratings related to the provisioning of activity information or the provisioning of location information in Step 3 by selection the option *I do not understand this statement*. However, when asked explicitly in Step 4 if they understood the ratings, only 50% of the people confirmed they understood, and 25% were not sure. We believe this result shows that users intuitively are able to rate providers for the different trust aspects but can not clearly explain the meaning of their ratings.

Our survey results also show that 25% of the survey participants found it difficult to rate the providers and 70% of the people would like to receive recommendations about the trust belief ratings from other people. This indicates that to our survey participants, trust recommendations are a desired feature of trust management models.

***Recommendation of Providers***

In step 4 of our survey, we compared the output of our trust management mechanism with the survey participants' choices in step 2. In the survey we refer to the output of our trust management mechanism as our *recommendation* of location provider, activity provider, identity provider, and service provider. We used the goal from step 2 and the trust ratings of the survey participants from step 3 as input for our trust management mechanism.

The survey participants could read their choices, see the recommendation, and read the reasoning behind the recommendation of our trust management mechanism. For each choice by the survey participant that did not correspond to the recommendation, the survey participant could choose to accept our recommendation and change his/her mind.

Our evaluation questions with respect to the usability, usefulness, and validity of our trust management mechanism were:

- How many people agree with the recommendation of our trust management mechanism with respect to the choice of location, activity, identity, and service provider?
- Why do people agree or disagree with the recommendation of our trust management mechanism?
- How many people understand the reasoning of our trust management mechanism?
- How many people think the recommendation of our trust management mechanism is useful?
- How many people would accept automatic selection of providers?

Figure 5-6 shows for how many of the survey participants the *location provider* they chose was in agreement with the recommendation of our trust management mechanism. 87% of the survey participants either agreed or decided to agree with the recommendation of our trust management mechanism with respect to the location information provider. Excluding those survey participants who did not care about the location provider (28 people), 40% of the survey participants chose exactly the location provider recommended by our trust management mechanism.

Figure 5-7 shows for how many of the survey participants the *activity provider* they chose was in agreement with the recommendation of our trust management mechanism. 82% of the survey participants either agreed or decided to agree with the recommendation of our trust

management mechanism with respect to the activity information provider. Excluding those survey participants who did not care about the activity provider (15 people), 57% of the survey participants chose exactly the activity provider recommended by our trust management mechanism.

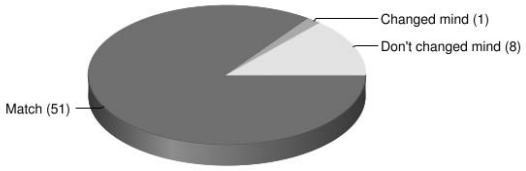



Figure *5-6* Pie chart showing for how many of the survey participants the location provider they chose was in agreement with the recommendation of our trust management mechanism

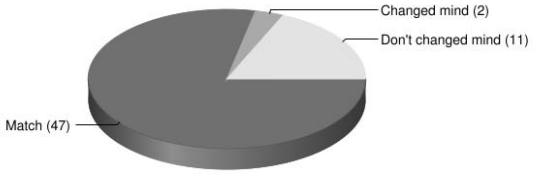



Figure *5-7* Pie chart showing for how many of the survey participants the activity provider they chose was in agreement with the recommendation of our trust management mechanism

Figure *5-8* shows for how many of the survey participants the *identity provider* they chose was in agreement with the recommendation of our trust management mechanism. 89% of the survey participants either agreed or decided to agree with the recommendation of our trust management mechanism with respect to the identity provider. Excluding those survey participants who did not care about the identity provider (13 people), 67% of the survey participants chose exactly the identity provider recommended by our trust management mechanism.

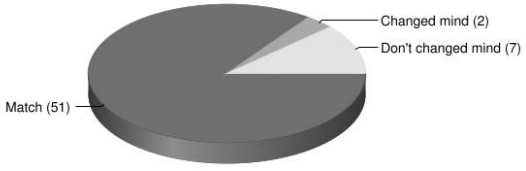



Figure *5-8* Pie chart showing for how many of the survey participants the identity provider they chose was in agreement with the recommendation of our trust management mechanism

Figure *5-9* shows for how many of the survey participants the *service provider* they chose was in agreement with the recommendation of our trust management mechanism. 82% of the survey participants either agreed or decided to agree with the recommendation of our trust management mechanism with respect to the service provider. Excluding those survey participants who did not care about the identity provider (11

people), 64% of the survey participants chose exactly the service provider recommended by our trust management mechanism.

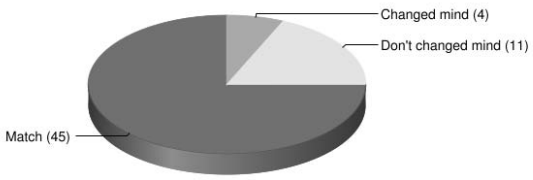



Figure *5-9* Pie chart showing for how many of the survey participants the service provider they chose was in agreement with the recommendation of our trust management mechanism

Our user survey included one open-ended question, asking the survey participants why they agree or disagree with our recommendation of providers. The reasons for disagreement were misunderstandings related to the trustworthiness values of the information providers and privacy issues. The survey participants were confused by the fact that our mechanism does not evaluate the quality level of the activity and location information but only the trustworthiness aspects. Furthermore, the survey participants were concerned with privacy issues related to other people being able to see their location and activity information, and not with the privacy issues related to the handling of their activity and location information by the context information and service providers. Some people also re-evaluated their ratings based on the recommendation of our trust management mechanism.

People who did not agree with the output of our trust management mechanism stated that:

- It's Friend Radar, I do not use my company as an identity provider because I would like to keep my company separate from my friends. Would it have been Colleague Radar, then I would have chosen my company as an identity provider;
- Didn't realize that my answers were such that I believe that my mobile phone operator is more trustworthy than Google. I think they are equally trustworthy, so I stick with my choice;
- I have some misunderstanding regarding the term trustworthy;
- If you can present more evidence that the recommended provider is better, I have no problem to change it. Otherwise it does not matter who is the provider;
- Are you trustworthy? In other words, why should I believe you, on what evidence is your recommendation based?

People who agreed with the output of our trust management mechanism stated that:

- You are the expert; I am a beginner. So I accept your recommendation;
- Since some answers were not consistent, I trust your recommendation in detecting that.

When asked explicitly if they understood the recommendation of our trust management mechanism 72% of the survey participants answered that they understood the recommendation of our trust management mechanism, 54% answered that they thought the recommendation was useful, and 57% answered that they would not accept automatic selection of providers.

Our user survey included two closed questions and two open-ended questions asking the survey participants if they agree with the reasoning of our trust management mechanism when privacy is more important for them than the functionality of the service. Considering the responses for the open-ended questions, we believe that the two questions were ambiguous and not well explained, because the survey participants mostly answered that they could not understand these questions. For this reason, we excluded these four questions from our survey results[2].

### 5.2.2 Context-Based Privacy Management

Our evaluation questions with respect to the validity of our context-based privacy management, privacy preview, and privacy quiz mechanisms were:

- How many users find it necessary to have context-based privacy preferences?
- How many users find it necessary to specify their own context-based privacy preferences?
- In what situations would people like to specify their personalized context-based privacy preferences?
- Do people find the mechanisms of privacy preview and privacy quiz useful?

From the survey results we conclude that for 70% of the survey participants, static groups of people (e.g. manually defined list of friends) are not enough to specify their privacy preferences, 77% of the people think their privacy preferences are different depending on their context situation, and 72% believe they would like to personalize their privacy preferences considering their context situation. Our user survey included one open-ended question asking the survey participants to provide examples of situations they would like to allow access to their location and activity information according to personalized privacy preferences. The following context-based privacy preferences examples were reported by the survey participants:

- At home, sports events, public spot somewhere, during weekends;
- Let my family know where I am;
- My wife permanent access to my location;
- Inform my wife when traveling home from work;

[2] The excluded questions are 6, 7, 8 and 9, see Appendix C, Step 4.

- When meeting friends downtown for drinks, Saturday evening when I am out in the city center;
- Close by friends when we are both in an unusual place (e.g., both on holidays in Paris);
- During vacations/traveling/trips (has my plane landed yet), traveling together in groups with several cars, flying somewhere together, in a train;
- When I am in a lecture during weekdays allow access to my location to my friends and family;
- Inform all the parents of the hockey team I am coaching where we are on the trip home so they know when we are expected back to pick up the kids;
- If I am chronically ill or otherwise need medical attention out of the hospital I can imagine that I would allow medical personnel to know my location. They would have to improve their privacy awareness though;
- Emergencies, for example, when I am in a train crash;
- Special situation, for example when I am on a trip with colleagues;
- At work, conference, in a meeting, in the classroom;
- I am at work and I have a meeting with some colleague;
- Only work-related, e.g., to colleagues and my secretary during working hours;
- Allow for all colleagues my working hours activities;
- Access to my location is always allowed to my colleagues during work hours;
- To my boss and colleagues during working hours when I am actually sitting at my desk;
- In the office to let my colleagues know whether I am available for a short meeting or not;
- Traffic jams (so people can see why I am late for an appointment);
- If I were working doing delivery of products;
- At a conference to find friends/colleagues.

In our survey, we took the opportunity to ask two questions about the privacy preview and privacy quiz mechanisms described in Chapter 4. These mechanisms are not major contributions of our work. However, they are innovative concepts that potentially can be used in future implementations of privacy control mechanisms. Our results show that 97% of the survey participants thought that the privacy preview is useful and 70% thought that the privacy quiz is useful. One of the survey participants answered in the open-ended questions that the privacy quiz should pop up at regular intervals, to make sure that the user of the service agrees with his/her privacy preferences.

## 5.3 Summary and Final Considerations

In this chapter, we evaluate whether the survey participants understood the concepts in our trust model with aspect-specific trust relationships and whether their choices of providers and their trust beliefs validate the recommendation of our trust mechanism. Except for the work done by Antifakos et al. [5], which investigates if the usability of a context-aware service increases when the system displays a confidence value for situation detection to the service users, we are not aware of any other user study in the area of trust management for context-aware service platforms.

Due to the nature of our survey we are not able to draw statistically valid conclusions with respect to our sub-goals. However, we can get an indication of the usefulness, usability, and validity of the contributions of this thesis from our survey participants' point of view. The following list summarizes our results:

- Most of the participants (75%) thought that privacy is more important than the functionality of the service;
- Around 95% of the survey participants understood: (a) the different roles in a context-aware service platform and understood the difference when choosing the context provider, identity provider, and service provider; (b) the different trust aspects and were able to provide their trust beliefs regarding these aspects with respect to the different providers. This result gives an indication that our trust management model is usable from a user perspective because it includes concepts that could be understood;
- For 85% of the survey participants, the recommendation of providers from our trust management mechanism based on their goal and trust beliefs was in agreement with their choices. This result indicates that the reasoning of our trust management mechanism is in accordance with the reasoning of the survey participants and validates our mechanism for the *Friend Radar* context-aware service scenario;
- Around 70% of the survey participants thought that static groups are not enough and they need personalized context-based privacy support to manage their privacy preferences. This result confirms the relevance of our mechanism to support context-based personalized policy management.

The open-ended questions in our survey provided rich feedback about future extensions that could be added to our trust management model and mechanisms, and to our context-based trust and privacy management approach. Through the open-ended questions we were able to learn examples of goals, other than privacy and functionality of the service, that the users have when using a context-aware service, and examples of context-based privacy preferences and situations that the users are interested on.

Examples of trust aspects that could be added are cost, acessibility, coverage, integration/compatibility with other services and devices, and safety. Examples of context-based policies include preferences for sport and free time activities, close family members, friends, and health emergency situations.

Asking the users might be different than actually observing their behavior when using a context-aware service. Other trust and privacy user studies show that users do not behave in the way they state they would in a given situation [90]. We acknowledge that firmer indications on the usefulness, usability, and validity of our research can only be obtained by building and deploying a context-aware system such as Friend Radar; however, such work was beyond the scope of this thesis.

Chapter 6

# Conclusions

This chapter summarizes the contributions of this thesis together with a critical analysis of these contributions. The critical analysis of the contributions leads to the identification of open issues that require further investigation. This chapter is further structured as follows. Section 6.1 summarizes our major research contributions. Section 6.2 describes future work.

## 6.1 Major Research Contributions

The major research contributions of this thesis are selection mechanisms and a new approach for context-based policy management to support service users and service providers in managing the trade-off between privacy protection and context-based service adaptation. The user survey we conducted also contributes with increased knowledge about the trust and privacy management requirements of context-aware service platforms, which helps to better understand the problems addressed by this thesis. The following list describes our major contributions depicted in Figure *6-1*. The arrows in the figure depict how the stakeholders and the contributions are related to each other and to the new models and concepts proposed by us.

1. **Trust-based Selection of Service Providers**: a trust-based selection mechanism that support users of context-aware services in selecting trustworthy entities to interact with. Our mechanisms use as input the users' goals and trust beliefs focusing on trust aspects related to identity provisioning, context information provisioning, privacy enforcement, and context-aware service provisioning. This mechanism is novel and original because we identify trust relationships, design a trust management model, and propose a trust management mechanism that address the aspect-specific trust dependen-

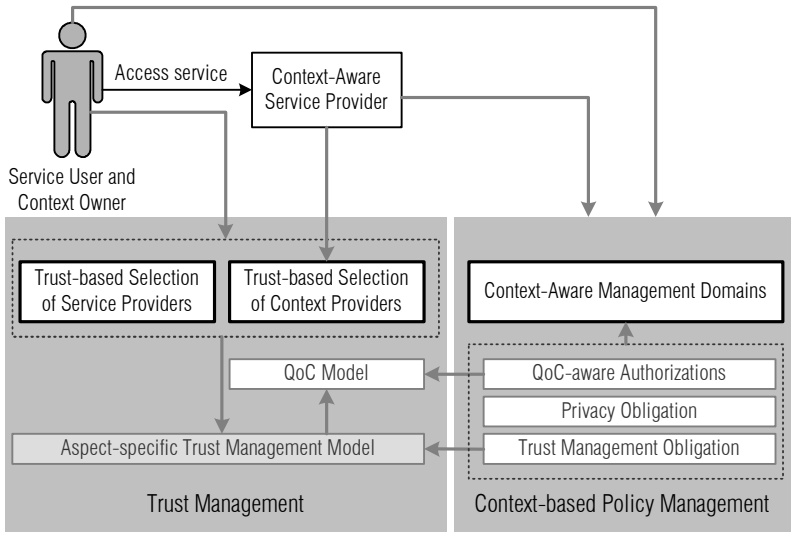



Figure *6-1* Research contributions of this thesis

cies between the different stakeholders in an overall trustworthiness evaluation of a context-aware service considering the trade-off addressed by this thesis;

2. **Trust-based Selection of Context Providers**: a trust-based selection mechanism that support context-aware service providers in selecting trustworthy context information providers considering a particular trustworthiness value, context owner, and QoC level. A minor contribution is our QoC model, which does not include trustworthiness as an attribute of the context information. Trustworthiness is defined in our trust management model as a trust aspect of the context provider with respect to its capabilities of providing context information. This mechanism is novel and original because it clearly distinguishes trustworthiness from the other QoC attributes and show in practice how to combine trust management and QoC models;
3. **Context-based Policy Management**: a policy management concept called Context-Aware Management Domain (CAMD) that uses context information as input for the policy management task and supports users and system administrators in the specification of personalized context-based QoC-aware authorizations, privacy obligations, and trust management obligation policies. This contribution is novel and original because we are the first to provide a generic framework for context-based policies that allows integrated management of context-based authorizations and obligations. With our CAMD concept we are able to express context-based policies with temporal constraints because we consider in our model the concept

of context situations. At the best of our knowledge there is no other policy languages that provides this expressiveness.

The following subsections detail our contributions and discuss the validation aspects including the user survey we conducted.

### 6.1.1 Trust-based Selection of Service Providers

Our trust management model supports the quantification of trust, taking uncertainty into account, for each of the different trust aspects we have identified in the analysis of our target context-aware service platform. Our trust model addresses trust aspects related to identity provisioning, privacy enforcement, context information provisioning, and context-aware service provisioning. Existing trust models address at most one of these trust aspects in isolation, and are therefore unable to handle the dependencies we have identified in the evaluation of a context-aware service provider trustworthiness. With our trust management model we are able to calculate an overall trustworthiness of a context-aware service provider which considers the trustworthiness of all entities the service providers depends on into account.

We have identified the dependencies between these trust aspects and proposed a mechanism to combine the trust values related to the different trust aspects in order to evaluate the resulting trust users have in a context-aware service provider. We address two different resulting trust evaluation approaches considering the user goals of privacy enforcement and service provisioning. These two user goals are related to the trade-off between privacy protection and context-based service adaptation.

Our trust model is extensible because additional trust aspects can be added and trust assessment mechanisms considering other goals can be specified using the concepts we propose in our trust management model. New trust assessment mechanisms benefit from the trust quantification considering uncertainty and Subjective Logic operators that are part of our trust management model.

We showed that our trust management model is technically feasible and demonstrated through our case study, proof-of-concept prototype, and user survey how to assist context-aware service users in selecting trustworthy context-aware service providers, context information providers, and identity providers. Our user survey contributes to knowledge improvement with respect to trust and privacy goals and to identify other requirements of context-aware service users. With our user survey we confirm the trade-off we address in this thesis and also learned additional user goals and trust aspects that are relevant for users. Our survey results give strong indications that our trust management model and mechanism are usable and useful. Most of the survey participants understood and agreed with the trust concepts we support in our model, and agreed with the output and reasoning of our trust management mechanisms.

### 6.1.2 Trust-based Selection of Context Providers

The QoC model proposed by this thesis is based on state-of-the-art QoC models and contributes to a better understanding of these models using as a reference quality concepts and vocabulary defined in a ISO standard for metrology. Our QoC model clearly distinguishes the QoC concepts and directly benefits developers of context-aware service platforms because it shows the difference between the QoC concepts and how they can be applied in practice.

We also showed how to apply our QoC model in practice in a trust management mechanism to support context-aware service providers in managing QoC and trustworthiness values. The objective of our mechanism is to support the selection of trustworthy context providers and consequently to increase the reliability capability of service providers for context-based service adaptation. We demonstrated the technical feasibility of our QoC management model through a case study and proof-of-concept prototype implementation that shows how our trust model and mechanisms can be used to define QoC attributes, to manage the trustworthiness values, and to select context providers that return numeric values of ambient temperature.

### 6.1.3 Context-based Policy Management

The novel concept of CAMDs introduced by this thesis allows generic and dynamic context-based policy management for context-aware service platforms. In comparison to existing context-based policy management solutions, CAMDs are more flexible because they are not limited to a specific policy management area such as access control, trust management obligations, or privacy obligations. Furthermore, the CAMD concept provides an abstraction for specifying obligation policies based on context situation events, which is not currently supported by any context-based policy management solution we are aware of in the state of the art. We demonstrated the technical feasibility of our CAMD concept through two case studies and prototype implementations using the Ponder2 and XACML policy languages to manage authorization and obligations.

Based on our case studies we proved that our CAMD concept is technically feasible and expressive to support the specification of personalized context-based QoC-aware authorizations, privacy obligations, and trust management obligation policies. We also showed that our CAMD concept is generic and can be applied in policy management frameworks other than Ponder2 because we have successfully applied our CAMD concept in a case study using an XACML implementation. In our case studies we learned innovative ideas to support users of a context-aware service platform in reviewing their privacy preferences by means of a privacy preview

and privacy quiz mechanism. Our user survey shows that personalized context-based privacy preferences are required and that the survey participants were able to understand how authorization policies regarding their context information could be managed using a CAMD.

## 6.2 Future Research

Context-based management of user goals, quality of context, and trust recommendations

One possible direction is to specialize our CAMD mechanism to support policies related to user goals, Quality of Context, and the trust recommendation process. Context could be used to dynamically adapt the user goals because in certain context situations users may change their goal when they need better context-based service adaptation (e.g., to send an ambulance to their current trustworthy location). Context could also be included in our trust mechanisms because a context provider might be able to provide a better QoC level depending on the context situation; for instance, a GPS receiver only works when used in an open environment. Context situations can be used to determine suitable target entities to request recommendations from, which we believe could allow anonymous and still useful exchange of trust recommendations. These types of policies are challenging because they require a complex and expressive conceptual model. Users of the context-aware services need to understand these concepts in order to allow user-based personalization of these type of policies.

Concrete metrics and mechanism for obtaining trust belief values

Another area for future research is the investigation of practical approaches to obtain trust belief values for the different trust aspects. In our examples, case studies, and prototype implementations, we arbitrarily defined initial trust values for each trust aspect in order to illustrate the usefulness of our trust model and mechanisms. Examples of concrete metrics for trust calculation include reputation mechanisms, trust values that map context conditions to a specific trust value, statistical approaches to allow the inference of trust values from analysis of past experiences, and strong trust support using hardware mechanisms such as the Trusted Computing Platform (TCP). The challenge in the specification of concrete metrics is the validation of the trust values and a meaningful interpretation of the trust value semantics for different entities when trust recommendations are exchanged.

Trustworthiness feedback mechanisms

In our QoC management mechanism, we propose using feedback from the context-aware service users to manage the trustworthiness values for the context providers. A possible future research direction is to verify whether users are more likely to report negative experiences than positive ones. It is possible that users are more likely to provide negative experiences because people usually do not complain when things

are working well. If this assumption is true, the feedback mechanism proposed by this thesis should be adapted to increase the trust values of the context information providers every time context information is provided, instead of increasing the trust values only when positive feedback is received.

Overlap analysis in context-based policies

In the case study of the Colleague Radar service, we implemented a mechanism for detecting overlap in context-based authorization policies. The problem with overlaps is that the service users might specify contradicting or redundant policies. For example, a policy that allows everybody access to a person's location information is contradictory to a policy denying access to the person's location to their co-workers. The challenge is in the detection of policy contradictions and in the specification of languages to allow specification of contradiction resolution strategies, for example, using defeasible reasoning.

For context-based policies, the detection of overlapping policies and contradicting actions is even more challenging because overlap in context situations might be detected only at execution time. One example is a policy that denies access to a patient's location information to everybody who is outside of a hospital, and another policy that allows access to the patient's location information to doctors. When doctors are outside of the hospital, should access be allowed or denied? The main problem we foresee is the specification of adequate runtime mechanisms to identify and address conflicting actions of overlapping context-based policies.

Generic goal-based trust management mechanisms for service-oriented architectures

The trust management model and mechanisms proposed in this thesis are specific for context-aware service platforms, and take into account the specific sets of goals of the service users and service providers. For Service-Oriented Architectures (SOAs) in general, other stakeholder goals, policies, and trust requirements are needed. Despite the general application of our contributions, we envision that a generalization of our trust management mechanism using rules that could be easily adapted for different service scenarios and business compositions would be of great value. Some of these extensions are already proposed by us in work developed after this thesis [76].

Appendix A

# Health Service PonderTalk Policies

*Listing A-1* Bootstrap CAMD Manager

```
// Import the managed object code
domain        := root load: "Domain".
event         := root load: "EventTemplate".
authorization := root load: "AuthorisationPolicy".
obligation    := root load: "ObligationPolicy".

// Create the default domain structure
root
at: "event"         put: domain create;
at: "policy"        put: domain create;
at: "factory"       put: domain create.

// Add factories to domain
root/factory
at: "domain"        put: domain;
at: "event"         put: event;
at: "obligation"    put: obligation;
at: "authorization" put: authorization.
```

*Listing A-2* Bootstrap health service managed objects and domains

```
// Create variables
domain        := root/factory/domain.
event         := root/factory/event.
authorization := root/factory/authorization.
obligation    := root/factory/obligation.
policies      := root/policy.

patient       := root load: "health.Patient".
caregiver     := root load: "health.Caregiver".
doctor        := root load: "health.Doctor".

root/factory
at: "patient"        put: patient;
at: "caregiver"      put: caregiver;
at: "doctor"         put: doctor.

// Basic health domain structure
root at: "Health_domain" put: domain create.
root/Health_domain
at: "Patients"          put: domain create;
at: "Caregivers"        put: domain create;
at: "Doctors"           put: domain create;
at: "Imminent_seizures" put: domain create.
```

*Listing A-3* Define health service events received from CHP

```
// Define events
EnterTrueSeizure  :=
  event create: #( "patient" "startTime" "nearbyAvailableCaregivers" ).
EnterFalseSeizure :=
  event create: #( "patient" ).
CaregiverAcceptedToHelp := event create: #( "patient" "caregiver" ).
root/event
at: "EnterTrueSeizure"        put: EnterTrueSeizure;
at: "EnterFalseSeizure"       put: EnterFalseSeizure;
at: "CaregiverAcceptedToHelp" put: CaregiverAcceptedToHelp.
```

*Listing A-4* Helper function to create a domain by name and return reference

```
createDomain := [ :target :name |
target at: name put: (domain create).
target resolve: name.
].
```

*Listing A-5* When EnterTrueSeizure event is received create the CAMD structure

```
camdCreate := obligation create.
camdCreate
event: EnterTrueSeizure;
condition: [1==1];
action: [
    // Event parameters
:patient :startTime :nearbyAvailableCaregivers |

patientName := patient getName.
camdName := "Seizure_patient_" + patientName.
root print: "Creating CAMD [" + camdName + "]".
camd := createDomain value: root/Health_domain/Imminent_seizures
value: camdName.

root print: " - adding policies domain".
policies := createDomain value: camd value: "Policies".

root print: " - adding caregivers that accepted to help".
domCaregiversAcceptedToHelp := createDomain value: camd
value: "Caregivers_that_accepted_to_help".

root print: " - adding nearby and available caregivers".
domNearbyAvailableCaregivers := createDomain value: camd
value: "Nearby_and_available_caregivers".
nearbyAvailableCaregivers do: [ :caregiver |
domNearbyAvailableCaregivers at: (caregiver getName) put: caregiver
    ].

root print: " - Adding start time".
domStartTime := createDomain value: camd value: "Start_time".
domStartTime at: startTime put: startTime.

root print: " - Adding patient".
domPatient := createDomain value: camd value: "Patient".
domPatient at: patientName put: patient.

// CAMD policies go here, see next listing

];
active: true.
root/policy at: "camdCreate" put: camdCreate.
```

*Listing A-6* Associate QoC-aware authorization policies with a CAMD

```
// Allow access to patient location at city level
// QoC-aware authorization
// Subject Focus
auth1 := authorization
  subject: domNearbyAvailableCaregivers
action:  "getLocation:" target:  domPatient
focus:   "s".
auth1 reqcondition: [:qoc | qoc == "CityLevel"].
auth1 active: true.
policies at: "auth1" put: auth1.
// Target Focus
auth2 := authorization
  subject: domNearbyAvailableCaregivers
action:  "getLocation:" target:  domPatient
focus:   "t".
auth2 reqcondition: [:qoc | qoc == "CityLevel"].
auth2 active: true.
policies at: "auth2" put: auth2.

// Allow access to patient location at street level
// QoC-aware authorization
// Subject Focus
auth3 := authorization
  subject: domCaregiversAcceptedToHelp
action:  "getLocation:" target:  domPatient
focus:   "s".
auth3 reqcondition: [:qoc | qoc == "StreetLevel"].
auth3 active: true.
policies at: "auth3" put: auth3.
// Target Focus
auth4 := authorization
  subject: domCaregiversAcceptedToHelp
action:  "getLocation:" target:  domPatient
focus:   "t".
auth4 reqcondition: [:qoc | qoc == "StreetLevel"].
auth4 active: true.
policies at: "auth4" put: auth4.

// Allow access to patient health data
// Authorization
// Subject Focus
auth5 := authorization
  subject: domCaregiversAcceptedToHelp
action:  "getHealthData" target:  domPatient
focus:   "s".
auth5 reqcondition: [1==1].
auth5 active: true.
policies at: "auth5" put: auth5.
// Target Focus
auth6 := authorization
  subject: domCaregiversAcceptedToHelp
action:  "getHealthData" target:  domPatient
focus:   "t".
auth6 reqcondition: [1==1].
auth6 active: true.
policies at: "auth6" put: auth6.
```

*Listing A-7* When CaregiverAcceptedToHelp event is received update CAMD structure and execute trust management obligation

```
camdUpdate := obligation create.
camdUpdate
event: CaregiverAcceptedToHelp;
condition: [1==1];
action: [

// Event parameters
:patient :caregiver |

patientName := patient getName.
camdName := "Seizure_patient_" + patientName.
root print: "Updating CAMD [" + camdName + "]".
camd := root/Health_domain/Imminent_seizures resolve: camdName.

root print: " - updating caregivers that accepted to help".
domCaregiversAcceptedToHelp := camd resolve: "Caregivers_that_accepted_to_help".

caregiverName := (caregiver getName).
domCaregiversAcceptedToHelp at: caregiverName put: caregiver.

root print: "Increasing trust in caregiver: " + caregiverName.
/root/trust/trustProvider increaseTrust: caregiver.

];
active: true.
root/policy at: "camdUpdate" put: camdUpdate.
```

*Listing A-8* When EnterFalseSeizure is received delete CAMD structure revoking access to location and health data, and fulfill privacy obligations

```
camdDelete := obligation create.
camdDelete
event: EnterFalseSeizure;
condition: [1==1];
action: [

// Event parameters
:patient |

patientName := patient getName.
camdName := "Seizure_patient_" + patientName.
root print: "Deleting CAMD [" + camdName + "]".
camd := root/Health_domain/Imminent_seizures resolve: camdName.

domCaregiversAcceptedToHelp := camd resolve: "Caregivers_that_accepted_to_help".

caregivers := domCaregiversAcceptedToHelp listObjects.
caregivers do: [ :value |
value deleteHealthData: (patient getName).
value deleteLocation: (patient getName).
].

(camd resolve: "Policies/auth1") active: false.
(camd resolve: "Policies/auth2") active: false.
(camd resolve: "Policies/auth3") active: false.
(camd resolve: "Policies/auth4") active: false.
(camd resolve: "Policies/auth5") active: false.
(camd resolve: "Policies/auth6") active: false.

camd removeAll.

root/Health_domain/Imminent_seizures remove: camdName.
];
active: true.
root/policy at: "camdDelete" put: camdDelete.
```

*Listing A-9* Create health simulated scenario and generate events to test CAMD deployment

```
patient   := root/factory/patient.
caregiver := root/factory/caregiver.
doctor    := root/factory/doctor.

// Patients
root/Health_domain/Patients
at: "Ricardo_Neisse" put: (patient create: "Ricardo Neisse").

// Caregivers
root/Health_domain/Caregivers
at: "Maarten_Wegdam"       put: (caregiver create: "Maarten Wegdam");
at: "Marten_van_Sinderen"   put: (caregiver create: "Marten van Sinderen").

ricardo := root/Health_domain/Patients/Ricardo_Neisse.
maarten := root/Health_domain/Caregivers/Maarten_Wegdam.
marten  := root/Health_domain/Caregivers/Marten_van_Sinderen.

event   := root/event/EnterTrueSeizure.
event create: #( ricardo "2011-09-15 15:00" #( maarten marten )).

event   := root/event/CaregiverAcceptedToHelp.
event create: #( ricardo maarten ).

event   := root/event/EnterFalseSeizure.
event create: #( ricardo ).
```

Appendix B

# Office Service XACML Policies

In order to increase the readability of the XML policies the Uniform Resource Name (URN) *urn:oasis:names:tc:xacml:1.0* and *urn:oasis:names:tc:xacml:2.0* have been replaced by the shorter versions *u1* and *u2*.

*Listing B-1* The outer policy set that contains all XACML policies

```
<?xml version="1.0" encoding="UTF-8"?>
<PolicySet xmlns="u2:policy:schema:cd-01"
  PolicyCombiningAlgId="u1:policy-combining-algorithm:first-applicable"
  PolicySetId="DomainPolicy">
  <Description>Top-level PolicySet</Description>
  <Target/>

  <!-- All policies go here -->

</PolicySet>
```

*Listing B-2* Default policy that deny access in case none of the other policies permit access

```
<Policy PolicyId="DenyEverybodyElse"
  RuleCombiningAlgId="u1:rule-combining-algorithm:first-applicable">
  <Target/>
  <Rule Effect="Deny" RuleId="DenyEverybodyElse"/>
</Policy>
</PolicySet>
```

*Listing B-3* Policy that prohibits all actions if the user privacy preference *Appear offline* is selected

```
<Policy PolicyId="CR_INVISIBLE"
  RuleCombiningAlgId="u1:rule-combining-algorithm:first-applicable">
  <Description>I closed my eyes, nobody can see me!</Description>
  <Target>
    <Resources>
      <Resource>
        <ResourceMatch
          MatchId="u1:function:string-equal">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string">true</AttributeValue>
          <ResourceAttributeDesignator
            AttributeId="urn:cmf:resource:preference:ColleagueRadar:CR_INVISIBLE"
            MustBePresent="false"
            DataType="http://www.w3.org/2001/XMLSchema#string"/>
        </ResourceMatch>
      </Resource>
    </Resources>
  </Target>
  <Rule
    Effect="Deny"
    RuleId="CR_INVISIBLE"/>
</Policy>
```

*Listing B-4* Policy that checks if subject and resource id/domain are the same, allowing an entity access to its own context information

```
<Policy PolicyId="AllowOwner"
  RuleCombiningAlgId="u1:rule-combining-algorithm:first-applicable">
  <Target/>
  <Rule
    Effect="Permit"
    RuleId="AllowOwner">
    <Condition
      FunctionId="u1:function:and">
      <Apply
        FunctionId="u1:function:string-equal">
        <Apply
          FunctionId="u1:function:string-one-and-only">
          <ResourceAttributeDesignator
            AttributeId="u1:resource:resource-id"
            MustBePresent="false"
            DataType="http://www.w3.org/2001/XMLSchema#string"/>
        </Apply>
        <Apply
          FunctionId="u1:function:string-one-and-only">
          <SubjectAttributeDesignator
            AttributeId="u1:subject:subject-id"
            MustBePresent="false"
            DataType="http://www.w3.org/2001/XMLSchema#string"/>
        </Apply>
      </Apply>
      <Apply
        FunctionId="u1:function:string-equal">
        <Apply
          FunctionId="u1:function:string-one-and-only">
          <ResourceAttributeDesignator
            AttributeId="urn:cmf:resource:resource-domain"
            MustBePresent="false"
            DataType="http://www.w3.org/2001/XMLSchema#string"/>
        </Apply>
        <Apply
          FunctionId="u1:function:string-one-and-only">
          <SubjectAttributeDesignator
            AttributeId="urn:cmf:subject:subject-domain"
            MustBePresent="false"
            DataType="http://www.w3.org/2001/XMLSchema#string"/>
        </Apply>
      </Apply>
    </Condition>
  </Rule>
</Policy>
```

*Listing B-5* Policy that allow access for all subjects if the first domain policy of the main privacy control GUI *Everybody can see me* is selected

```
<Policy PolicyId="DomainPolicy_1"
  RuleCombiningAlgId="u1:rule-combining-algorithm:first-applicable">
  <Description>Everybody is allowed to see me</Description>
  <Target>
    <Resources>
      <Resource>
        <ResourceMatch
          MatchId="u1:function:string-equal">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string">true</AttributeValue>
          <ResourceAttributeDesignator
            AttributeId="urn:cmf:resource:preference:ColleagueRadar:DomainPolicy_1"
            MustBePresent="false"
            DataType="http://www.w3.org/2001/XMLSchema#string"/>
        </ResourceMatch>
      </Resource>
    </Resources>
  </Target>
  <Rule
    Effect="Permit"
    RuleId="DomainPolicy_1"/>
</Policy>
```

*Listing B-6* Policy that allow access only to buddies if the second domain policy of the main privacy control GUI *Only buddies can see me* is selected

```
<Policy PolicyId="DomainPolicy_2"
  RuleCombiningAlgId="u1:rule-combining-algorithm:first-applicable">
  <Description>Only buddies are allowed to see me</Description>
  <Target>
    <Resources>
      <Resource>
        <ResourceMatch
          MatchId="u1:function:string-equal">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string">true</AttributeValue>
          <ResourceAttributeDesignator
            AttributeId="urn:cmf:resource:preference:ColleagueRadar:DomainPolicy_2"
            MustBePresent="false"
            DataType="http://www.w3.org/2001/XMLSchema#string"/>
        </ResourceMatch>
      </Resource>
    </Resources>
    <Subjects>
      <Subject>
        <SubjectMatch
          MatchId="u1:function:string-equal">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string">Buddies</AttributeValue>
          <SubjectAttributeDesignator
            AttributeId="urn:cmf:subject:subject-group"
            DataType="http://www.w3.org/2001/XMLSchema#string"
            MustBePresent="false"/>
        </SubjectMatch>
      </Subject>
    </Subjects>
  </Target>
  <Rule
    Effect="Permit"
    RuleId="DomainPolicy_2"/>
</Policy>
```

*Listing B-7* Policy that allow access to all context information to everybody when resource is in the CAMD *inside the building* and only if this option is selected in the basic privacy GUI

```
<Policy PolicyId="DomainPolicy_3"
  RuleCombiningAlgId="u1:rule-combining-algorithm:permit-overrides">
  <Description>DomainPolicy_3: Allow everybody access; if I'm in the
    office</Description>
  <Target>
    <Resources>
      <Resource>
        <ResourceMatch
          MatchId="u1:function:string-equal">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string">true</AttributeValue>
          <ResourceAttributeDesignator
            AttributeId="urn:cmf:resource:preference:ColleagueRadar:DomainPolicy_3"
            MustBePresent="false"
            DataType="http://www.w3.org/2001/XMLSchema#string"/>
        </ResourceMatch>
      </Resource>
    </Resources>
  </Target>
  <Rule
    Effect="Permit"
    RuleId="DomainPolicy_3">
    <Condition
      FunctionId="u1:function:boolean-is-in">
      <AttributeValue
        DataType="http://www.w3.org/2001/XMLSchema#boolean">true</AttributeValue>
      <ResourceAttributeDesignator
        AttributeId="urn:cmf:resource:camd:inbuilding"
        DataType="http://www.w3.org/2001/XMLSchema#boolean"/>
    </Condition>
  </Rule>
</Policy>
```

*Listing B-8* Policy that allow access to all context information to buddies when resource is in the CAMD *inside the building* and only if this option is selected in the basic privacy GUI

```
<Policy PolicyId="DomainPolicy_4"
  RuleCombiningAlgId="u1:rule-combining-algorithm:permit-overrides">
  <Description>DomainPolicy_4: Allow Buddies access; if I'm in the
    office</Description>
  <Target>
    <Resources>
      <Resource>
        <ResourceMatch
          MatchId="u1:function:string-equal">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string">true</AttributeValue>
          <ResourceAttributeDesignator
            AttributeId="urn:cmf:resource:preference:ColleagueRadar:DomainPolicy_4"
            MustBePresent="false"
            DataType="http://www.w3.org/2001/XMLSchema#string"/>
        </ResourceMatch>
      </Resource>
    </Resources>
    <Subjects>
      <Subject>
        <SubjectMatch
          MatchId="u1:function:string-equal">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string">Buddies</AttributeValue>
          <SubjectAttributeDesignator
            AttributeId="urn:cmf:subject:subject-group"
            DataType="http://www.w3.org/2001/XMLSchema#string"
            MustBePresent="false"/>
        </SubjectMatch>
      </Subject>
    </Subjects>
  </Target>
  <Rule
    Effect="Permit"
    RuleId="DomainPolicy_4">
    <Condition
      FunctionId="u1:function:boolean-is-in">
      <AttributeValue
        DataType="http://www.w3.org/2001/XMLSchema#boolean">true</AttributeValue>
      <ResourceAttributeDesignator
        AttributeId="urn:cmf:resource:camd:inbuilding"
        DataType="http://www.w3.org/2001/XMLSchema#boolean"/>
    </Condition>
  </Rule>
</Policy>
```

*Listing B-9* Policy that allow everybody access to all context information if the user (resource) is not inside of the building during office hours (from 09:00 until 17:00) and this option is selected by the user in the basic privacy GUI

```
<Policy PolicyId="DomainPolicy_5"
  RuleCombiningAlgId="u1:rule-combining-algorithm:permit-overrides">
  <Description>DomainPolicy_5: Allow Everybody access; if I'm outside
    the office; during office hours</Description>
  <Target>
    <Resources>
      <Resource>
        <ResourceMatch
          MatchId="u1:function:string-equal">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string">true</AttributeValue>
          <ResourceAttributeDesignator
            AttributeId="urn:cmf:resource:preference:ColleagueRadar:DomainPolicy_5"
            MustBePresent="false"
            DataType="http://www.w3.org/2001/XMLSchema#string"/>
        </ResourceMatch>
      </Resource>
    </Resources>
  </Target>
  <Rule
    Effect="Permit"
    RuleId="DomainPolicy_5">
    <Condition
      FunctionId="u1:function:and">
      <Apply
        FunctionId="u1:function:and">
        <Apply
          FunctionId="u1:function:time-greater-than-or-equal">
          <Apply
            FunctionId="u1:function:time-one-and-only">
            <EnvironmentAttributeDesignator
              AttributeId="u1:environment:current-time"
              DataType="http://www.w3.org/2001/XMLSchema#time"/>
          </Apply>
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#time">09:00:00</AttributeValue>
        </Apply>
        <Apply
          FunctionId="u1:function:time-less-than-or-equal">
          <Apply
            FunctionId="u1:function:time-one-and-only">
            <EnvironmentAttributeDesignator
              AttributeId="u1:environment:current-time"
              DataType="http://www.w3.org/2001/XMLSchema#time"/>
          </Apply>
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#time">17:00:00</AttributeValue>
        </Apply>
      </Apply>
      <Apply
        FunctionId="u1:function:boolean-is-in">
        <AttributeValue
          DataType="http://www.w3.org/2001/XMLSchema#boolean">false</AttributeValue>
        <ResourceAttributeDesignator
          AttributeId="urn:cmf:resource:camd:inbuilding"
          DataType="http://www.w3.org/2001/XMLSchema#boolean"/>
      </Apply>
    </Condition>
  </Rule>
</Policy>
```

*Listing B-10* Policy that allow buddies access to all context information if the user (resource) is not inside of the building during office hours (from 09:00 until 17:00) and this option is selected by the user in the basic privacy GUI

```
<Policy PolicyId="DomainPolicy_6"
  RuleCombiningAlgId="u1:rule-combining-algorithm:permit-overrides">
  <Description>DomainPolicy_6: Allow Buddies access; if I'm outside
    the office; during office hours</Description>
  <Target>
    <Resources>
      <Resource>
        <ResourceMatch
          MatchId="u1:function:string-equal">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string">true</AttributeValue>
          <ResourceAttributeDesignator
            AttributeId="urn:cmf:resource:preference:ColleagueRadar:DomainPolicy_6"
            MustBePresent="false"
            DataType="http://www.w3.org/2001/XMLSchema#string"/>
        </ResourceMatch>
      </Resource>
    </Resources>
    <Subjects>
      <Subject>
        <SubjectMatch
          MatchId="u1:function:string-equal">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string">Buddies</AttributeValue>
          <SubjectAttributeDesignator
            AttributeId="urn:cmf:subject:subject-group"
            DataType="http://www.w3.org/2001/XMLSchema#string"
            MustBePresent="false"/>
        </SubjectMatch>
      </Subject>
    </Subjects>
  </Target>
  <Rule
    Effect="Permit"
    RuleId="DomainPolicy_6">
    <Condition
      FunctionId="u1:function:and">
      <Apply
        FunctionId="u1:function:and">
        <Apply
          FunctionId="u1:function:time-greater-than-or-equal">
          <Apply
            FunctionId="u1:function:time-one-and-only">
            <EnvironmentAttributeDesignator
              AttributeId="u1:environment:current-time"
              DataType="http://www.w3.org/2001/XMLSchema#time"/>
          </Apply>
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#time">09:00:00</AttributeValue>
        </Apply>
        <Apply
          FunctionId="u1:function:time-less-than-or-equal">
          <Apply
            FunctionId="u1:function:time-one-and-only">
            <EnvironmentAttributeDesignator
              AttributeId="u1:environment:current-time"
              DataType="http://www.w3.org/2001/XMLSchema#time"/>
          </Apply>
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#time">17:00:00</AttributeValue>
        </Apply>
      </Apply>
      <Apply
        FunctionId="u1:function:boolean-is-in">
        <AttributeValue
          DataType="http://www.w3.org/2001/XMLSchema#boolean">false</AttributeValue>
        <ResourceAttributeDesignator
          AttributeId="urn:cmf:resource:camd:inbuilding"
          DataType="http://www.w3.org/2001/XMLSchema#boolean"/>
      </Apply>
    </Condition>
  </Rule>
</Policy>
```

*Listing B-11* Policy that allow everybody access to all context information if the user (resource) is not inside of the building outside office hours (NOT from 09:00 until 17:00) and this option is selected by the user in the basic privacy GUI

```
<Policy PolicyId="DomainPolicy_7"
  RuleCombiningAlgId="u1:rule-combining-algorithm:permit-overrides">
  <Description>DomainPolicy_7: Allow Everybody access; if I'm outside
    the office; during non office hours</Description>
  <Target>
    <Resources>
      <Resource>
        <ResourceMatch
          MatchId="u1:function:string-equal">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string">true</AttributeValue>
          <ResourceAttributeDesignator
            AttributeId="urn:cmf:resource:preference:ColleagueRadar:DomainPolicy_7"
            MustBePresent="false"
            DataType="http://www.w3.org/2001/XMLSchema#string"/>
        </ResourceMatch>
      </Resource>
    </Resources>
  </Target>
  <Rule
    Effect="Permit"
    RuleId="DomainPolicy_7">
    <Condition
      FunctionId="u1:function:and">
      <Apply
        FunctionId="u1:function:not">
        <Apply
          FunctionId="u1:function:and">
          <Apply
            FunctionId="u1:function:time-greater-than-or-equal">
            <Apply
              FunctionId="u1:function:time-one-and-only">
              <EnvironmentAttributeDesignator
                AttributeId="u1:environment:current-time"
                DataType="http://www.w3.org/2001/XMLSchema#time"/>
            </Apply>
            <AttributeValue
              DataType="http://www.w3.org/2001/XMLSchema#time">09:00:00</AttributeValue>
          </Apply>
          <Apply
            FunctionId="u1:function:time-less-than-or-equal">
            <Apply
              FunctionId="u1:function:time-one-and-only">
              <EnvironmentAttributeDesignator
                AttributeId="u1:environment:current-time"
                DataType="http://www.w3.org/2001/XMLSchema#time"/>
            </Apply>
            <AttributeValue
              DataType="http://www.w3.org/2001/XMLSchema#time">17:00:00</AttributeValue>
          </Apply>
        </Apply>
      </Apply>
      <Apply
        FunctionId="u1:function:boolean-is-in">
        <AttributeValue
          DataType="http://www.w3.org/2001/XMLSchema#boolean">false</AttributeValue>
        <ResourceAttributeDesignator
          AttributeId="urn:cmf:resource:camd:inbuilding"
          DataType="http://www.w3.org/2001/XMLSchema#boolean"/>
      </Apply>
    </Condition>
  </Rule>
</Policy>
```

*Listing B-12* Policy that allow budies access to all context information if the user (resource) is not inside of the building outside office hours (NOT from 09:00 until 17:00) and this option is selected by the user in the basic privacy GUI

```
<Policy PolicyId="DomainPolicy_8"
  RuleCombiningAlgId="u1:rule-combining-algorithm:permit-overrides">
  <Description>DomainPolicy_8: Allow Buddies access; if I'm outside
    the office; during non office hours</Description>
  <Target>
    <Resources>
      <Resource>
        <ResourceMatch MatchId="u1:function:string-equal">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string">true</AttributeValue>
          <ResourceAttributeDesignator
            AttributeId="urn:cmf:resource:preference:ColleagueRadar:DomainPolicy_8"
            MustBePresent="false"
            DataType="http://www.w3.org/2001/XMLSchema#string"/>
        </ResourceMatch>
      </Resource>
    </Resources>
    <Subjects>
      <Subject>
        <SubjectMatch MatchId="u1:function:string-equal">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string">Buddies</AttributeValue>
          <SubjectAttributeDesignator AttributeId="urn:cmf:subject:subject-group"
            DataType="http://www.w3.org/2001/XMLSchema#string"
            MustBePresent="false"/>
        </SubjectMatch>
      </Subject>
    </Subjects>
  </Target>
  <Rule Effect="Permit" RuleId="DomainPolicy_7">
    <Condition
      FunctionId="u1:function:and">
      <Apply FunctionId="u1:function:not">
        <Apply FunctionId="u1:function:and">
          <Apply FunctionId="u1:function:time-greater-than-or-equal">
            <Apply FunctionId="u1:function:time-one-and-only">
              <EnvironmentAttributeDesignator
                AttributeId="u1:environment:current-time"
                DataType="http://www.w3.org/2001/XMLSchema#time"/>
            </Apply>
            <AttributeValue
              DataType="http://www.w3.org/2001/XMLSchema#time">09:00:00</AttributeValue>
          </Apply>
          <Apply
            FunctionId="u1:function:time-less-than-or-equal">
            <Apply
              FunctionId="u1:function:time-one-and-only">
              <EnvironmentAttributeDesignator
                AttributeId="u1:environment:current-time"
                DataType="http://www.w3.org/2001/XMLSchema#time"/>
            </Apply>
            <AttributeValue
              DataType="http://www.w3.org/2001/XMLSchema#time">17:00:00</AttributeValue>
          </Apply>
        </Apply>
      </Apply>
      <Apply
        FunctionId="u1:function:boolean-is-in">
        <AttributeValue
          DataType="http://www.w3.org/2001/XMLSchema#boolean">false</AttributeValue>
        <ResourceAttributeDesignator
          AttributeId="urn:cmf:resource:camd:inbuilding"
          DataType="http://www.w3.org/2001/XMLSchema#boolean"/>
      </Apply>
    </Condition>
  </Rule>
</Policy>
```

*Listing B-13* Example template policy for advanced privacy control screen. Access to user (resource) location (QoC = floor of the building) is allowed to buddies (subject)

```
<Policy PolicyId="UserPolicy_8"
  RuleCombiningAlgId="u1:rule-combining-algorithm:permit-overrides">
  <Description>When I'm outside the building access to my location (floor
    of the building) is allowed to my buddies</Description>
  <PolicyDefaults>
    <XPathVersion>http://www.w3.org/TR/1999/Rec-xpath-19991116</XPathVersion>
  </PolicyDefaults>
  <Target>
    <Subjects>
      <Subject>
        <SubjectMatch
          MatchId="u1:function:string-equal">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string">Buddies</AttributeValue>
          <SubjectAttributeDesignator
            AttributeId="urn:cmf:subject:subject-group"
            DataType="http://www.w3.org/2001/XMLSchema#string"/>
        </SubjectMatch>
      </Subject>
    </Subjects>
    <Resources>
      <Resource>
        <ResourceMatch MatchId="u1:function:string-equal">
          <AttributeValue
            DataType="http://www.w3.org/2001/XMLSchema#string">bob@ti.nl</AttributeValue>
          <ResourceAttributeDesignator AttributeId="u1:resource:resource-id"
            DataType="http://www.w3.org/2001/XMLSchema#string"/>
        </ResourceMatch>
      </Resource>
    </Resources>
    <Actions>
      <Action>
        <ResourceMatch MatchId="u1:function:string-equal">
          <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
            http://amigo/owl/ContextTransport.owl#CombinedUserLocation
</AttributeValue>
          <ActionAttributeDesignator AttributeId="u1:action:action-id"
            DataType="http://www.w3.org/2001/XMLSchema#string"/>
        </ResourceMatch>
      </Action>
    </Actions>
  </Target>
  <Rule RuleId="RuleId" Effect="Permit">
    <Description></Description>
    <Target>
      <Subjects><AnySubject/></Subjects>
      <Resources><AnyResource/></Resources>
      <Actions><AnyAction/></Actions>
    </Target>
    <Condition FunctionId="u1:function:boolean-equal">
      <Apply FunctionId="u1:function:boolean-one-and-only">
        <ResourceAttributeDesignator AttributeId="urn:cmf:resource:camd:outside"
          DataType="http://www.w3.org/2001/XMLSchema#boolean"/>
      </Apply>
      <AttributeValue
        DataType="http://www.w3.org/2001/XMLSchema#boolean">true</AttributeValue>
    </Condition>
  </Rule>
  <Obligations>
    <Obligation
      ObligationId =
        "http://amigo/owl/ContextTransport.owl#CombinedUserLocation"
      FulfillOn="Permit">
      <AttributeAssignment AttributeId="urn:awareness:names:cmf:qoc:1.0:precision"
        DataType="http://www.w3.org/2001/XMLSchema#integer">2</AttributeAssignment>
    </Obligation>
  </Obligations>
</Policy>
```

Appendix C

# User Survey about the Friend Radar Service

***Introduction***

The objective of this survey is to learn your choices and opinions with respect to the trust and privacy issues of a service called Friend Radar. The Friend Radar service runs on your mobile phone and allows you and your friends to visualize location and activity information about each other.

The completion time of this survey is around 15 minutes and all the data provided by you will be kept private and will be used only for research purposes.

The survey has 4 main steps:

1. In the first step we present a detailed description of the Friend Radar service to help you understand what the Friend Radar service offers to you;
2. In the second step we ask you to answer 5 questions about the Friend Radar service, for instance, what is important for you when using the service;
3. In the third step we ask you to rate your level of agreement or disagreement with a list of 10 statements related to the Friend Radar service;
4. In the fourth step we evaluate your choices and ask you to answer around 20 final questions about the Friend Radar service.

*IMPORTANT: Because the Friend Radar service is not yet available, please answer the questions to your best knowledge imagining that you are a user of this service and of the technologies related to it. Please do not use the browser's back button; if you want to restart the survey close the window/tab and open the address of the survey again.*

Before we begin please fill in the following fields:

– E-mail: (optional) (If you provide your email we will send you a summary of our research results)

- University/company: (optional)
- Age:
- Gender: Male Female
- Are you from a computer science/engineering or similar background: Yes No
- What is your country of residence?
- Select from the list the technologies you are familiar with: (optional) Smartphone or PDA, GPS navigation, Wireless networks, Outlook calendar, Skype.

***Step 1 - Detailed Description of the Friend Radar Service***

The Friend Radar service allows your friends and other people to see location and activity information about you. You specify who should be authorized to access your location and on your mobile phone.

This screen is displayed to you when you start the Friend Radar service on your mobile phone (Figure *C-1* left). After the service has started you are required to select your identity provider and fill in your username and password, which represent your digital identity. The Friend Radar accepts digital identities provided by Google, Skype, or the university or company you work for (Figure *C-2* right).


Figure *C-1* Start-up screen and your digital identity

After you are authenticated by your identity provider the Friend Radar service presents a list of your friends and options to visualize their current location or activity (Figure *C-2* left). If you choose to visualize the location you see the map with your friends' location. If you prefer you can see the location and activity of other people by checking the option Show everybody. The information you will be able to see depends on the other people's privacy preferences (Figure *C-2* center). If you choose to visualize the activity you see a list with your friends' present activity (Figure *C-2* right).


Figure *C-2* Your friends' lists, your friends' locations, and your friends' activities

In the Friend Radar service you can specify who is authorized to access your information through your privacy preferences. You can allow everybody that is using the Friend Radar service access to your location or activity or only to your friends. If you prefer to specify in more details you can access more privacy options. You can also check what other people can see about you or take a privacy quiz (Figure *C-3* left). You can select from a list of options when your friends or other users of the Friend Radar service are allowed to access your location and activity. This list allows you to specify in which situation other people can see your information. If you can not find in this list the privacy preference that you would like to set you can add a new one (Figure *C-3* center). When adding a new privacy preference you can specify when your friends or other people can see your location and activity information. For instance, you can specify that your friends can only see your location and activity when you are exercising in the evening if they are also exercising (Figure *C-3* right).

The privacy quiz allows you to check if you understand your privacy preferences by answering questions related to who is authorized to access your location and activity information (Figure *C-4* left). The Friend Radar service summarizes for you when your friends and other people can see your location and activity information (Figure *C-4* right).

***Step 2 - Your Choices***

Please answer the questions below with respect to the Friend Radar service:

1. When using the Friend Radar service, what is more important for you?
   – The protection of my privacy: I want my privacy to be protected even if my friends might not see my precise location and activity

Figure *C-3* Your privacy, more privacy options, and add new privacy preference


Figure *C-4* Take a privacy quiz and what can people see about you?


   - The functionality of the service provided: I want my friends to be able to see my precise location/activity even if my privacy might not be protected

2. When using the Friend Radar service, assuming that your location information could be retrieved from the sources below, from which source would you prefer that your location is retrieved?
   - Your location derived from the GSM cell you are connected to provided by your mobile phone operator
   - Your location derived from the wireless base station you are connected to provided by your university/company
   - It does not matter for me, both sources are fine
   - I do not understand the difference
3. When using the Friend Radar service, assuming that your activity information could be retrieved from the sources below, from which source would you prefer that your activity is retrieved?
   - Your activity derived from your calendar (e.g. Outlook) provided by your university/company

- Your activity detected through sensors (e.g. accelerometer) in your mobile phone provided by your mobile phone operator
- It does not matter for me, both sources are fine
- I do not understand the difference

4. When using the Friend Radar service, you are required to present a digital identity. Assuming that you own valid identities from the following identity providers, which one of the digital identities would you prefer to use?
   - Identity provided by your university/company
   - Identity provided by Skype
   - Identity provided by Google
   - It does not matter for me, all identities are fine
   - I do not understand the difference
5. Independently from the source of your location and activity, the Friend Radar service is available from different service providers. Which one of these service providers would you prefer to use?
   - Friend Radar service provided by your university/company
   - Friend Radar service provided by your mobile phone operator
   - Friend Radar service provided by Google
   - It does not matter for me, all service providers are fine
   - I do not understand the difference

***Step 3 - Your ratings***

Please read the statements bellow and select one of the options that best represent your opinion about what is stated. You should interpret the options as follow:

- Strongly agree: I have reasons to believe in the statement and I am very sure;
- Agree: I have reasons to believe in the statement but I am not sure;
- Don't know: I do not have reasons to believe or disbelieve;
- Disagree: I have reasons to disbelieve in the statements but I am not sure;
- Strongly disagree: I have reasons to disbelieve in the statements and I am very sure;
- I do not understand this statement: the statement is not clear to me.

Statements:

1. Do you believe that your mobile phone operator is trustworthy to provide your location derived from the GSM cell you are connected to?
2. Do you believe that your mobile phone operator is trustworthy to enforce your privacy preferences?
3. Do you believe that your university/company is trustworthy to provide your location derived from the wireless base station you are connected to?

4. Do you believe that your university/company is trustworthy to enforce your privacy preferences?
5. Do you believe that your university/company is trustworthy to provide your activity derived from your calendar (e.g. Outlook)?
6. Do you believe that your mobile phone operator is trustworthy to provide your activity detected through sensors (e.g. accelerometer) in your mobile phone?
7. Do you believe that Google is trustworthy to enforce your privacy preferences?
8. Do you believe that your university/company is trustworthy to identify you through your account information?
9. Do you believe that Skype is trustworthy to identify you through your account information?
10. Do you believe that Google is trustworthy to identify you through your account information?

***Step 4 - Evaluation of your choices and ratings***

Your choices in Step 2 were:

– <answer to item 1 Step 2> the protection of my privacy / the functionality of the service provided
– Your location provided by: <answer to item 2 Step 2>
– Your activity provided by: <answer to item 3 Step 2>
– Digital identity provided by <answer to item 4 Step 2>
– The Friend Radar service provided by <answer to item 5 Step 2>

Custom text according to the ratings from Step 3:

1. *The protection of my privacy* and privacy enforcement trust is unknown or trustworthy: because privacy is more important for you, and you trust that your privacy preferences will be enforced, we recommend you choose MORE trustworthy location, activity, and identity providers when using the Friend Radar service
2. *The protection of my privacy* and privacy enforcement trust is untrustworthy: because privacy is more important for you, and you DO NOT TRUST that your privacy preferences will be enforced, we recommend you choose LESS trustworthy location, activity, and identity providers when using the Friend Radar service
3. *The functionality of the service provided*: because the functionality of the Friend Radar service is more important for you, we recommend you choose MORE trustworthy location, activity, and identity providers when using this service

Below is our recommendation of choices with respect to the Friend Radar service taking into account your ratings from Step 3. A checkmark (Figure C-5) indicates that our recommendation is in agreement with what you have chosen, while a cross (Figure C-5) indicates that our

recommendation is different from what you have chosen. A smile (Figure *C-5*) indicates that, according to your choices, you do not care about or you do not understand this item.

- <location information provider>
  - To provide your location because you believe that they/it are/is MORE trustworthy to enforce your privacy preferences and to provide location information about you than the other entities
  - To provide your location because you believe that they/it are/is MORE trustworthy to enforce your privacy preferences and LESS trustworthy to provide location information about you than the other entities
- <activity information provider>
  - To provide your activity because you believe that they/it are/is MORE trustworthy to enforce your privacy preferences and to provide activity information about you than the other entities
  - To provide your location because you believe that they/it are/is MORE trustworthy to enforce your privacy preferences and LESS trustworthy to provide activity information about you than the other entities
- <identity provider>
  - To identify you because you believe that this entity is MORE trustworthy to identify you through your account information
  - To identify you because you believe that this entity is LESS trustworthy to identify you through your account information
- <service provider>
  - To provide the Friend Radar service because you believe that they are MORE trustworthy to enforce your privacy preferences than the other entities

Figure *C-5* Icons

Please answer the following questions (Definitely yes / Yes / Don't know / No / Definitely no):

1. <If result of trust management mechanism according to ratings in Step 3 does not match choices of the users from Step 2> Your choices do not match our recommendation. After reading our recommendation, would you change your choices according to it? Please select for which providers below you would accept our recommendation: (optional)
   - Location provider
   - Activity provider
   - Identity provider
   - Service provider

2. Why have you decided to change your choices and accept our recommendation or not to change your choices? Could you explain your reasons? (optional)
3. Do you understand how the Friend Radar service works?
4. Do you understand our recommendation with respect to the choices of location, activity, identity, and service providers?
5. Was the recommendation with respect to the choices of location, activity, identity, and service providers useful?
6. Do you agree that if you do not trust the location, activity, and service providers to enforce your privacy preferences, and your goal is to protect your privacy, you should choose the LESS trustworthy identity provider and the LESS trustworthy activity and location providers?
7. If you answered no or definitely no to the last question, could you explain your reasons for the answer? (optional)
8. Do you agree that if your goal is the functionality of the service you should choose the MORE trustworthy location and activity providers to provide location and activity information about you and the MORE trustworthy identity provider to identify you?
9. If you answered no or definitely no to the last question, could you explain your reasons for the answer? (optional)
10. Was it difficult to choose the location, activity, identity, and service providers for the Friend Radar service in Step 2?
11. Would you accept location, activity, identity, and service providers for the Friend Radar service to be automatically chosen for you considering what you believe is important when using this service, for example, your privacy or the functionality of the service?
12. Was it difficult to rate the location, activity, identity, and service providers for the Friend Radar service in Step 3?
13. Would you agree to receive recommendations from other people for the ratings of the location, activity, identity, and service providers for the Friend Radar service?
14. Do you think you completely understand the ratings from Step 3?
15. For the Friend Radar service, do you think goals other than privacy or service adaptation are important (e.g. cost of the service, availability, etc.)?
16. Could you provide us with examples of goals you believe should be considered when using the Friend Radar service? (optional)
17. Do you think only two privacy preferences for friends and everybody else are enough for you?
18. Do you think your privacy preferences are different depending on your situation as presented in the figure below?

19. Do you think you would like to personalize your privacy preferences based on your situation, your friends' situation, and the time of the day as presented in the figure below?
20. Could you describe examples of situations where you would like to allow access to your location and activity information? (optional)
21. Do you think the mechanism of privacy quiz is useful to help you verify if you understand your privacy preferences?
22. Do you think it is useful to see a summary of your privacy preferences as presented in the figure below?
23. Please write down any further consideration about this survey: (optional)

***Final Message***

Thank you for your collaboration!

As soon as we have our results published you will be informed by e-mail.

Appendix D

# Results of User Survey about the Friend Radar Service

## D.1 Survey Participants Profile

The charts presented in this section summarize the data we collected in our survey with respect to the profile of the participants.

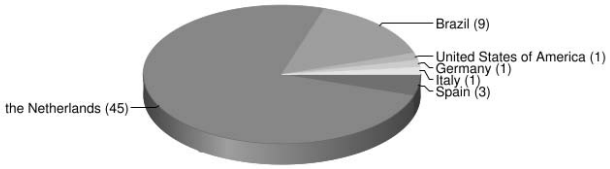



Figure *D-1* Pie chart summarizing the survey participants' country of residence


Figure *D-2* Gender of the survey participants

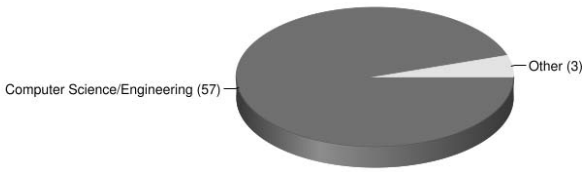



Figure *D-3* Survey participants who are from a computer science/engineering background

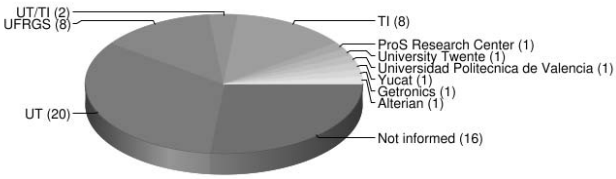



Figure *D-4* Pie chart showing the most popular institutions for the survey participants

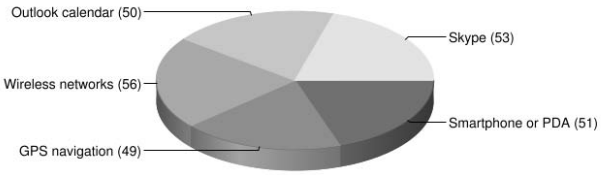



Figure *D-5* Pie chart showing the most popular technologies the survey participants are familiar with

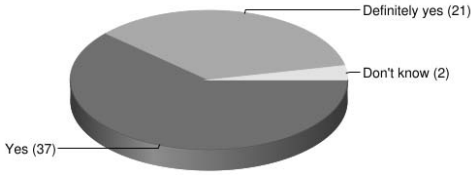



Figure *D-6* Pie chart showing the most popular answer of the survey participants when they were asked if they understand how the Friend Radar Service works

## D.2 Goals and Choices of Providers

The charts presented in this section summarize the data we collected in our survey with respect to the goals and choices of providers.

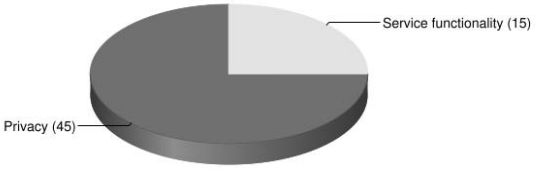



Figure *D-7* Pie chart showing the most popular goals for the survey participants

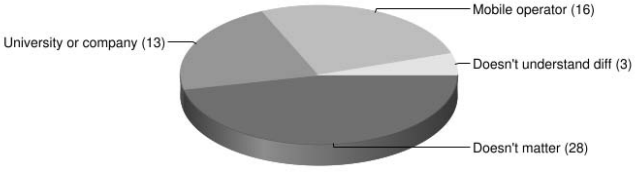



Figure *D-8* Pie chart showing the most popular location information providers for the survey participants

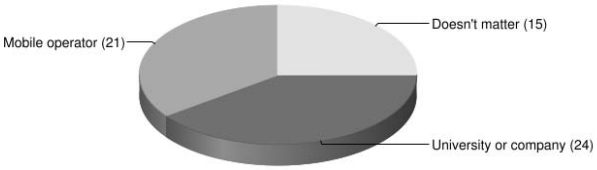



Figure *D-9* Pie chart showing the most popular activity information providers for the survey participants

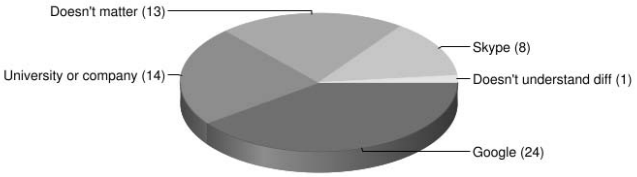



Figure *D-10* Pie chart showing the most popular identity providers for the survey participants

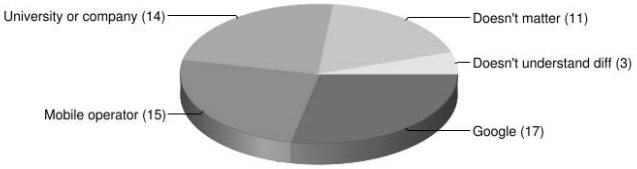



Figure *D-11* Pie chart showing the most popular service providers for the survey participants

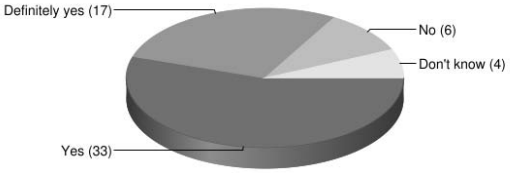



Figure *D-12* Pie chart showing how many of the survey participants think that goals other than privacy or functionality of the service are important

## D.3 Ratings or Trust Beliefs

The charts presented in this section summarize the data we collected in our survey with respect to the trust beliefs of the survey participants for Google, their mobile phone operator, their university or company, Skype, as well as usability and usefulness questions related to the survey participants' trust beliefs.

### D.3.1 Google

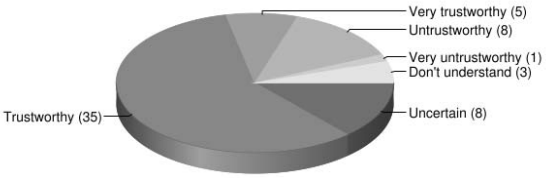



Figure *D-13* Pie chart showing how many of the survey participants trust Google to provide their digital identity

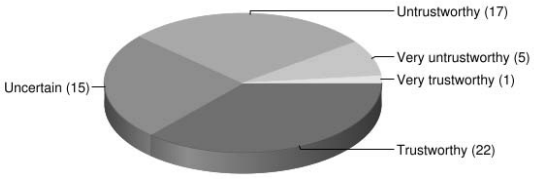



Figure *D-14* Pie chart showing how many of the survey participants trust Google to enforce their privacy preferences

### D.3.2 Mobile Phone Operator

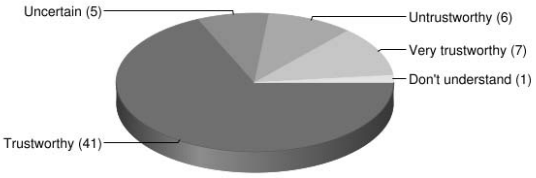



Figure *D-15* Pie chart showing how many of the survey participants trust their mobile phone operator to provide their location information

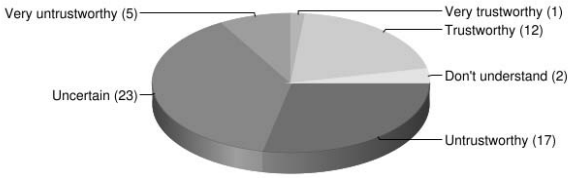



Figure *D-16* Pie chart showing how many of the survey participants trust their mobile phone operator to provide their activity information

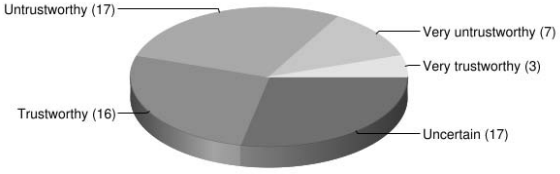



Figure *D-17* Pie chart showing how many of the survey participants trust their mobile phone operator to enforce their privacy preferences

### D.3.3 University or Company

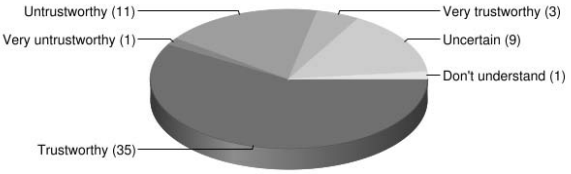



Figure *D-18* Pie chart showing how many of the survey participants trust their university or company to provide their location information

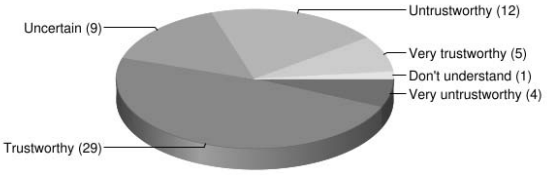



Figure *D-19* Pie chart showing how many of the survey participants trust their university or company to provide their activity information

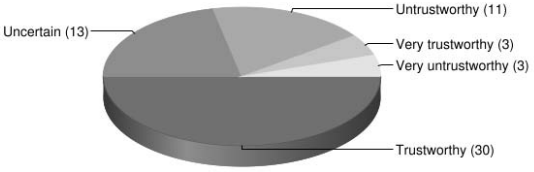



Figure *D-20* Pie chart showing how many of the survey participants trust their university or company to enforce their privacy preferences

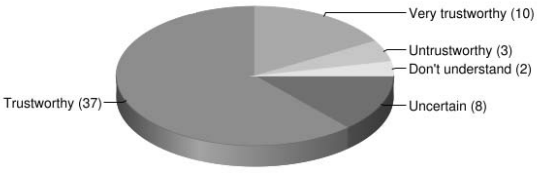



Figure *D-21* Pie chart showing how many of the survey participants trust their university or company to provide their digital identity

### D.3.4 Skype

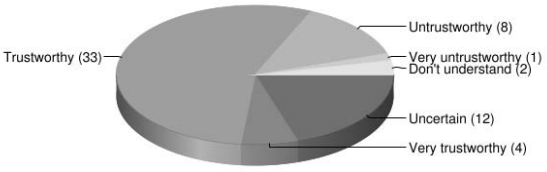



Figure *D-22* Pie chart showing how many of the survey participants trust Skype to provide their digital identity

### D.3.5 Usability and Usefulness

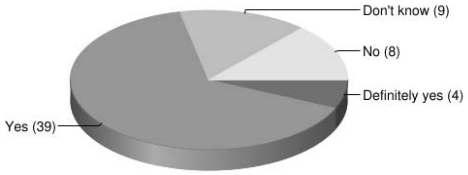



Figure *D-23* Pie chart showing how many of the survey participants would like to receive recommendations about the ratings

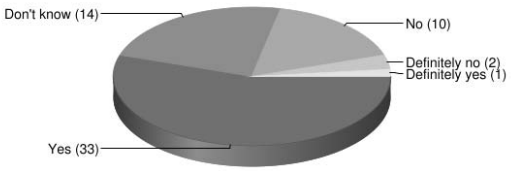



Figure *D-24* Pie chart showing how many of the survey participants understand the ratings

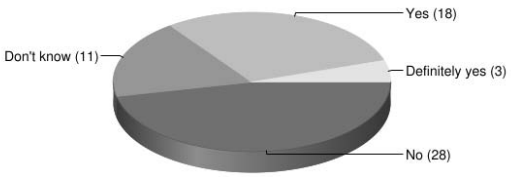



Figure *D-25* Pie chart showing how many of the survey participants find it difficult to rate the providers

## D.4 Validity of Trust Management Mechanism

The charts presented in this section summarize the data we collected in our survey with respect to the validity of our trust management mechanism.

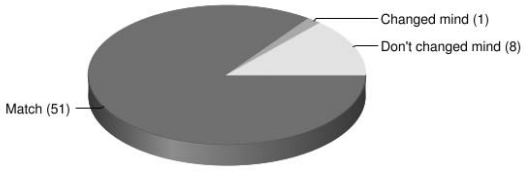



Figure *D-26* Pie chart showing for how many of the survey participants the location provider they chose was in agreement with the recommendation of our trust management mechanism

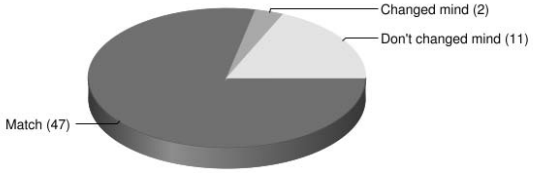



Figure *D-27* Pie chart showing for how many of the survey participants the activity provider they chose was in agreement with the recommendation of our trust management mechanism

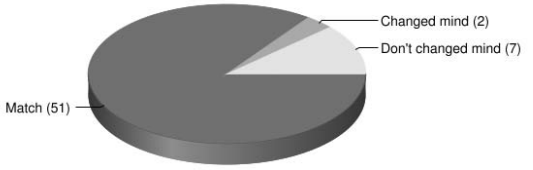



Figure *D-28* Pie chart showing for how many of the survey participants the identity provider they chose was in agreement with the recommendation of our trust management mechanism

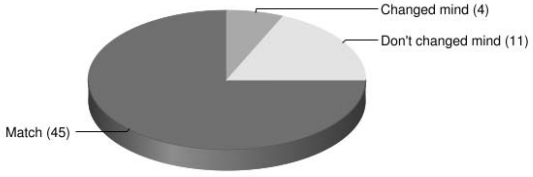



Figure *D-29* Pie chart showing for how many of the survey participants the service provider they chose was in agreement with the recommendation of our trust management mechanism

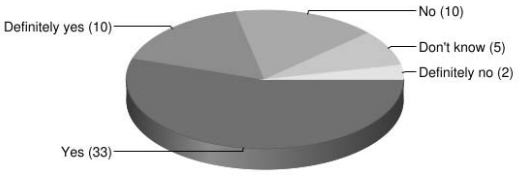



Figure *D-30* Pie chart showing how many of the survey participants understand the recommendation of our trust management mechanism

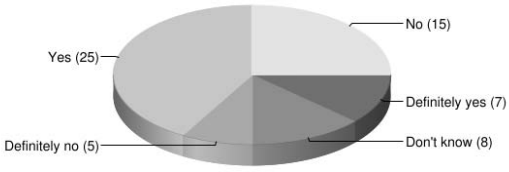



Figure *D-31* Pie chart showing how many of the survey participants think the recommendation of our trust management mechanism is useful

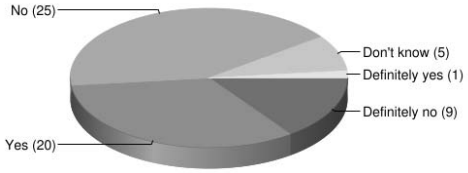



Figure *D-32* Pie chart showing how many of the survey participants would accept automatic selection of providers

## D.5 Context-Based Privacy Management

The charts presented in this section summarize the data we collected in our survey with respect to usability and usefulness of our context-based privacy management framework.

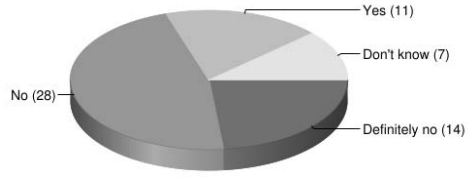



Figure *D-33* Pie chart showing how many of the survey participants think that static groups for friends and everybody else are not enough to specify their privacy preferences

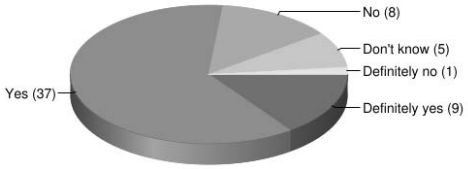



Figure *D-34* Pie chart showing how many of the survey participants think that their privacy preferences are different depending on their context situation

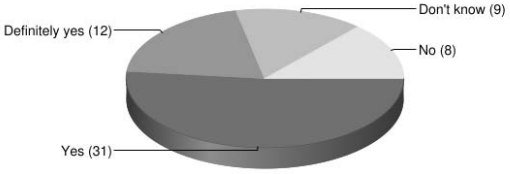



Figure *D-35* Pie chart showing how many of the survey participants think that they need personalized context-based privacy preferences

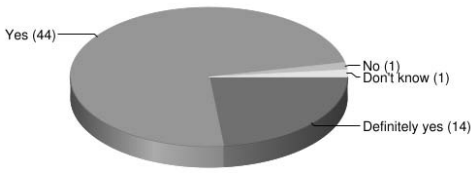



Figure *D-36* Pie chart showing how many of the survey participants think that the privacy preview mechanism is useful

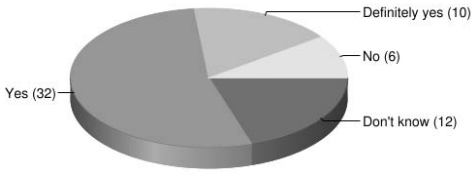



Figure *D-37* Pie chart showing how many of the survey participants think that the privacy quiz mechanism is useful

# Bibliography


[1] DoD 5200.28-STD. *Trusted Computer System Evaluation Criteria*. Dod Computer Security Center, December 1985.

[2] Alfarez Abdul-Rahman and Stephen Hailes. Supporting trust in virtual communities. In *HICSS '00: Proceedings of the 33rd Hawaii International Conference on System Sciences-Volume 6*, page 6007, Washington, DC, USA, 2000. IEEE Computer Society.

[3] F. Almenárez, A. Marín, C. Campo, and C. R. García. A pervasive trust management model for dynamic open environments. In *First Workshop on Pervasive Security and Trust in MobiQuitous 2004, Boston, USA*, 2004.

[4] Nicolas Anciaux, Harold van Heerde, Ling Feng, and Peter M.G. Apers. Implanting life-cycle privacy policies in a context database. Technical report, University of Twente, Enschede, The Netherlands, January 2006.

[5] Stavros Antifakos, Nicky Kern, Bernt Schiele, and Adrian Schwaninger. Towards improving trust in context-aware systems by displaying system confidence. In *MobileHCI '05: Proceedings of the 7th International Conference on Human Computer Interaction with Mobile Devices & Services*, pages 9–14, New York, NY, USA, 2005. ACM.

[6] P. Ashley, S. Hada, G. Karjoth, C. Powers, and M. Schunter. Enterprise privacy authorization language (epal 1.1) specification. Available at: `http://www.zurich.ibm.com/security/enterprise-privacy/epal`, 2003. IBM Research Report.

[7] Matthias Baldauf, Schahram Dustdar, and Florian Rosenberg. A survey on context-aware systems. *International Journal Ad Hoc Ubiquitous Computing*, 2:263–277, June 2007.

[8] David E. Bell and Leonard La Padula. Secure computer system: Unified exposition and multics interpretation, 1975. ESD-TR-75-306, ESD/AFSC, Hanscom AFB, Bedford, MA 01731 (1975).

[9] Bettina Berendt, Oliver Günther, and Sarah Spiekermann. Privacy in e-commerce: stated preferences vs. actual behavior. *Commun. ACM*, 48(4):101–106, 2005.

[10] C. Bettini, X. SeanWang, and S. Jajodia. Protecting privacy against location-based personal identification. In *Proceedings of 2nd VLDB Workshop on Secure Data Management (SDM)*, 2005.

[11] Claudio Bettini, Sushil Jajodia, X. Sean Wang, and Duminda Wijesekera. Provisions and obligations in policy management and security applications. In *Proceedings of the 28th international conference on Very Large Data Bases*, VLDB '02, pages 502–513. VLDB Endowment, 2002.

[12] Claudio Bettini, Sushil Jajodia, X. Sean Wang, and Duminda Wijesekera. Provisions and obligations in policy rule management. *J. Netw. Syst. Manage.*, 11:351–372, September 2003.

[13] K. J. Biba. Integrity considerations for secure computer systems. Technical report, MITRE Corp., 04 1977.

[14] M. Blaze, J. Feigenbaum, J. Ioannidis, and A. Keromytis. The KeyNote Trust-Management System Version 2. RFC 2704 (Informational), September 1999.

[15] Matt Blaze, Joan Feigenbaum, and Angelos D. Keromytis. Keynote: Trust management for public-key infrastructures. In *Infrastructures (Position Paper). Lecture Notes in Computer Science 1550*, pages 59–63, 1998.

[16] Matt Blaze, Joan Feigenbaum, and Jack Lacy. Decentralized trust management. In *Proceedings IEEE Symposium on Security and Privacy*, pages 164–173, 1996.

[17] Luiz Olavo Bonino da Silva Santos, Luís Ferreira Pires, and Marten J. van Sinderen. Architectural models for client interaction on service-oriented platforms. In *1st International Workshop on Architectures, Concepts and Technologies for Service Oriented Computing (ACT4SOC 2007)*, pages 19–27. INSTIC, July 2007.

[18] T. Buchholz, A. Küpper, and M. Schiffers. Quality of context information: What it is and why we need it. In *Proceedings of the 10th International Workshop of the HP OpenView University Association*

*(HPOVUA), Geneva, Switzerland*. Hewlett-Packard OpenView University Association, Jul. 2003.

[19] Jan Willem Bulle. Privacy & context aware systems, July 2008. Bachelor Thesis, Telematica Instituut and Hogeschool Rotterdam (in Dutch).

[20] F. Cabitza, M. Sarini, and B. Dal Seno. Jess rule engine. Available at: `http://herzberg.ca.sandia.gov/jess/`, 2008.

[21] CEI and ISO. *International vocabulary of basic and general terms in metrology (VIM)*. International Organization for Standardisation, 2004.

[22] Guanling Chen and David Kotz. A survey of context-aware mobile computing research. Technical report, Technical Report Dartmouth College, 2000.

[23] Harry Chen. *An Intelligent Broker Architecture for Pervasive Context-Aware Systems*. PhD thesis, University of Maryland, Baltimore County, December 2004.

[24] Rita Chen and William Yeager. Poblano: A distributed trust model for peer-to-peer networks. Technical report, Technical Report, Sun Microsystems, 2001.

[25] Y. Chu. REFEREE: trust management for Web applications. *Computer Networks and ISDN Systems*, September 1997.

[26] David D. Clark and David R. Wilson. A comparison of commercial and military computer security policies. *IEEE Symposium on Security and Privacy*, 0:184, 1987.

[27] Antonio Corradi, Rebecca Montanari, and Daniela Tibaldi. Context-based access control for ubiquitous service provisioning. In *COMPSAC '04: Proceedings of the 28th Annual International Computer Software and Applications Conference (COMPSAC'04)*, pages 444–451, Washington, DC, USA, 2004. IEEE Computer Society.

[28] Antonio Corradi, Rebecca Montanari, and Daniela Tibaldi. Context-driven adaptation of trust relationships in pervasive collaborative environments. In *SAINT-W '05: Proceedings of the 2005 Symposium on Applications and the Internet Workshops*, pages 178–181, Washington, DC, USA, 2005. IEEE Computer Society.

[29] Michael J. Covington, Wende Long, Srividhya Srinivasan, Anind K. Dev, Mustaque Ahamad, and Gregory D. Abowd. Securing context-aware applications using environment roles. In *SACMAT*

*'01: Proceedings of the Sixth ACM Symposium on Access Control Models and Technologies*, pages 10–20, New York, NY, USA, 2001. ACM.

[30] Lorrie Cranor, Brooks Dobbs, Serge Egelman, Giles Hogben, Jack Humphrey, Marc Langheinrich, Massimo Marchiori, Martin Presler-Marshall, Joseph Reagle, Matthias Schunter, David A. Stampley, and Rigo Wenning. The platform for privacy preferences 1.1 (p3p1.1) specification. Available at: `http://www.w3.org/TR/2006/NOTE-P3P11-20061113/`, 2006. W3C Working Group Note 13 November 2006.

[31] Marcin Czenko, Sandro Etalle, Dongyi Li, and William Winsborough. An introduction to the role based trust management framework rt. In *Foundations of Security Analysis and Design IV*, volume 4677 of *Lecture Notes in Computer Science*, pages 246–281. Springer Berlin / Heidelberg, 2007.

[32] N. Damianou, N. Dulay, E. Lupu, M. Sloman, and T. Tonouchi. Tools for domain-based policy management of distributed systems. *Network Operations and Management Symposium, 2002. NOMS 2002. 2002 IEEE/IFIP*, pages 203–217, 2002.

[33] Nicodemos Damianou, Naranker Dulay, Emil Lupu, and Morris Sloman. The ponder policy specification language. In *POLICY '01: Proceedings of the International Workshop on Policies for Distributed Systems and Networks*, pages 18–38, London, UK, 2001. Springer-Verlag.

[34] S. Daskapan, A. Ali Eldin, and R. Wagenaar. Trust in mobile context aware systems. In *International Business Information Management Conference (5th IBIMA), Dec. 2005, Cairo, Egypt*, 2005.

[35] IMDb The Internet Movie Database. Imdb - user ratings (votes). Available at: `http://www.imdb.com/help/\?ratings/`, 2008.

[36] Dorothy E. Denning. A lattice model of secure information flow. *Communications of the ACM*, 19, 1976.

[37] A. K. Dey. Understanding and using context. *Personal Ubiquitous Computing*, 5(1):4–7, 2001.

[38] NIST Computer Security Division. Role based access control - frequently asked questions. Available at: `http://csrc.nist.gov/groups/SNS/rbac/faq.html`, 2010.

[39] Patrícia Dockhorn Costa. *Architectural support for context-aware applications: from context models to services platforms*. PhD thesis, Univ. of Twente, December 2007.

[40] eBay. Trust and safety on ebay. Available at: `http://pages.ebay.com/help/newtoebay/resolving-concerns.html`, 2008.

[41] H. Batteram et al. Awareness scope and scenarios. Available at: `http://awareness.freeband.nl`, 2004. Deliverable D1.1.

[42] Seth Proctor et al. Sun's xacml implementation. Available at: `http://sunxacml.sourceforge.net`, 2009.

[43] Joan Feigenbaum. Overview of the at&t labs trust-management project. In *Proceedings of the 1998 Cambridge University Security Protocols International Workshop*. Springer, 1998.

[44] Guy G. Gable. Integrating case study and survey research methods: an example in information systems. *European Journal of Information Systems*, 3:112–126, 1994.

[45] Simon Godik and Tim Moses. extensible access control markup language (xacml) specification version 1.1. Available at: `http://www.oasis-open.org/committees/xacml/repository/cs-xacml-specification-1.1.pdf`, 2003.

[46] Dieter Gollmann. Why trust is bad for security. *Electron. Notes Theor. Comput. Sci.*, 157(3):3–9, 2006.

[47] Google. Google pagerank. Available at: `https://www.google.com`, 2008.

[48] T. Grandison and M. Sloman. A survey of trust in internet application. Available at: `http://citeseer.ist.psu.edu/grandison00survey.html`, 2000.

[49] Tyrone Grandison. *Trust Management for Internet Applications*. PhD thesis, Imperial College London, July 2003.

[50] Tyrone Grandison and Morris Sloman. Trust management tools for internet applications. In *Trust Management, First International Conference, iTrust 2003, Heraklion, Crete, Greece, May 28-30, 2002, Proceedings*, volume 2692 of *Lecture Notes in Computer Science*, pages 91–107. Springer, 2003.

[51] Trusted Computing Group. Architecture overview, version 1.4. Available at: `http://www.trustedcomputinggroup.org`, 2007.

[52] Trusted Computing Group. Tpm specification version 1.2 revision 103: Part 1 - design principles. Available at: `http://www.trustedcomputinggroup.org`, 2007.

[53] Trusted Computing Group. Tpm specification version 1.2 revision 103: Part 2 - structures. Available at: `http://www.trustedcomputinggroup.org`, 2007.

[54] Trusted Computing Group. Tpm specification version 1.2 revision 103: Part 3 - commands. Available at: `http://www.trustedcomputinggroup.org`, 2007.

[55] Karen Henricksen and Jadwiga Indulska. A software engineering framework for context-aware pervasive computing. *percom*, 00:77, 2004.

[56] Cristian Hesselman, Henk Eertink, and Martin Wibbels. Privacy-aware context discovery for next generation mobile services. In *Applications and the Internet Workshops, 2007. SAINT Workshops 2007. International Symposium on*, Jan 2007.

[57] Cristian Hesselman, Henk Eertink, Martin Wibbels, Kamran Sheikh, and Andrew Tokmakoff. Controlled disclosure of context information across ubiquitous computing domains. In *Proceedings of the 2008 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (sutc 2008)*, pages 98–105, Washington, DC, USA, 2008. IEEE Computer Society.

[58] M. Hilty, A. Pretschner, D. Basin, C. Schaefer, and T. Walter. A policy language for distributed usage control. In *Computer Security – ESORICS 2007*, Lecture Notes in Computer Science, pages 531–546. Springer Berlin / Heidelberg, 2008.

[59] Mario Hoffmann. User-centric identity management in open mobile environments. In *Workshop on Security and Privacy in Pervasive Computing*, Vienna, Austria, 2004.

[60] C. Huebscher and A. Mccann. An adaptive middleware framework for context-aware applications. *Personal Ubiquitous Comput.*, 10(1):12–20, December 2005.

[61] Markus C. Huebscher and Julie A. McCann. A learning model for trustworthiness of context-awareness services. In *PERCOMW '05: Proceedings of the Third IEEE International Conference on Pervasive Computing and Communications Workshops*, pages 120–124, Washington, DC, USA, 2005. IEEE Computer Society.

[62] R. J. Hulsebosch, A. H. Salden, M. S. Bargh, P. W. G. Ebben, and J. Reitsma. Context sensitive access control. In *SACMAT '05: Proceedings of the Tenth ACM Symposium on Access Control Models and Technologies*, pages 111–119, New York, NY, USA, 2005. ACM.

[63] IETF. Geographic location/privacy (geopriv) charter. Available at: `http://www.ietf.org/html.charters/geopriv-charter.html`, 2009.

[64] Merriam-Webster Inc. Merriam-webster online dictionary. Available at: `http://www.merriam-webster.com`, 2008.

[65] Sun Mycrosystems Inc. Jxta java peer-to-peer api. Available at: `http://www.jxta.org`, 2008.

[66] Maddy D. Janse, Peter Vink, Iris Soute, and Heleen Bol. Perceived privacy in ambient intelligent environments. *First International Workshop on Combining Context with Trust, Security and Privacy*, July 2007.

[67] Xiaodong Jiang and James A. Landay. Modeling privacy control in context-aware systems. *IEEE Pervasive Computing*, 01(3):59–63, 2002.

[68] Audun Jøsang. The right type of trust for distributed systems. In *NSPW '96: Proceedings of the 1996 Workshop on New Security Paradigms*, pages 119–131, 1996.

[69] Audun Jøsang. A logic for uncertain probabilities. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9(3):279–311, 2001.

[70] Audun Jøsang. Evidential reasoning with subjective logic. In *13th International Conference on Information Fusion*, 2010.

[71] Audun Jøsang. Subjective logic book (draft). Available at: `http://persons.unik.no/josang/papers/subjective_logic.pdf`, 2010.

[72] Audun Jøsang, John Fabre, Brian Hay, James Dalziel, and Simon Pope. Trust requirements in identity management. In *ACSW Frontiers '05: Proceedings of the 2005 Australasian workshop on Grid computing and e-research*, pages 99–108, Darlinghurst, Australia, Australia, 2005. Australian Computer Society, Inc.

[73] Audun Jøsang, Ross Hayward, and Simon Pope. Trust network analysis with subjective logic. In *ACSC '06: Proceedings of the 29th Australasian Computer Science Conference*, pages 85–94, Darlinghurst, Australia, Australia, 2006. Australian Computer Society, Inc.

[74] Audun Jøsang, Claudia Keser, and Theodosis Dimitrakos. Can we manage trust? In *iTrust 2005, Trust Management, Third International Conference, Paris, France*, pages 93–107, 2005.

[75] James B. D. Joshi, Rafae Bhatti, Elisa Bertino, and Arif Ghafoor. Access-control language for multidomain environments. *IEEE Internet Computing*, 8(6):40–50, 2004.

[76] Klaus Julisch, Philip Miseldine, Hoon Wei Lim, Nataliia Bielova, Stephan Neuhaus, Atle Refsdal, Domenico Presenza, Beatriz Gallego-Nicasio Crespo, Paul Kearney, David Sinclair, and Ricardo Neisse. D2.1.3: The master final protection and assessment model. *FP7 MASTER research project deliverable, Activity A2, Work Package WP2.1*, 2010.

[77] Sepandar D. Kamvar, Mario T. Schlosser, and Hector Garcia-Molina. The eigentrust algorithm for reputation management in p2p networks. In *Proceedings of the 12th international conference on World Wide Web*, pages 640–651, New York, NY, USA, 2003. ACM.

[78] P. Kolari, Li Ding, G. Shashidhara, A. Joshi, T. Finin, and L. Kagal. Enhancing web privacy protection through declarative policies. In *Sixth IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY)*, 2005.

[79] Pranam Kolari, Li Ding, Lalana Kagal, Shashidhara Ganjugunte, Anupam Joshi, and Tim Finin. Enhancing P3P Framework through Policies and Trust. Technical report, UMBC, September 2004.

[80] Jon Krakauer. *Into the Wild*. Anchor, 1997.

[81] M. Krause and I. Hochstatter. Challenges in modelling and using quality of context (qoc). In *Mobility Aware Technologies and Applications (MATA), Montreal, Canada*, volume 3744 of *Lecture Notes in Computer Science*, pages 324–333. Springer, October, 2005.

[82] Sven Lachmund, Thomas Walter, Laurent Gomez, Laurent Bussard, and Eddy Olk. Context-aware access control making access control decisions based on context information. *3rd Annual International Conference on Mobile and Ubiquitous Systems - Workshops*, pages 1–8, July 2006.

[83] Saadi Lahlou, Marc Langheinrich, and Carsten Röcker. Privacy and trust issues with invisible computers. *Commun. ACM*, 48(3):59–60, 2005.

[84] Marc Langheinrich. When trust does not compute – the role of trust in ubiquitous computing. In *Proceedings of Privacy Workshops of Ubicomp-03, 2003*, 2003.

[85] Wei Li, Fredrik Kilander, and Carl Gustaf Jansson. Toward a person-centric context aware system. In *Workshop on Requirements and Solutions for Pervasive Software Infrastructures, PERVASIVE*, 2006.

[86] R. C. Mayer, J. H. Davis, and D. F. Schoorman. An integrative model of organizational trust, 1995. The Academy of Management Review, Vol. 20, No. 3. (1995), pp. 709-734.

[87] C.J. McCollum, J.R. Messing, and L. Notargiacomo. Beyond the pale of mac and dac-defining new forms of access control. In *Proceedings., 1990 IEEE Computer Society Symposium on Research in Security and Privacy, 1990.*, 1990.

[88] D. Harrison Mcknight and Norman L. Chervany. The meanings of trust, 1996. Available at: `http://misrc.umn.edu/wpaper/WorkingPapers/9604.pdf`.

[89] D. Harrison McKnight, Larry L. Cummings, and Norman L. Chervany. Initial trust formation in new organizational relationships. Available at: `www.msu.edu/~mcknig26/InitialTrustAMR.pdf`, 1998. The Academy of Management Review, Vol. 23, No. 3 (Jul., 1998), pp. 473-490.

[90] Stig F. Mjølsnes and Marius Teigen. A survey on trust and privacy negotiability in the norwegian mobile telecom market. *Electronic Notes in Theoretical Computer Science*, 179:135–142, 2007.

[91] B. Moore, E. Ellesson, J. Strassner, and A. Westerinen. Policy Core Information Model – Version 1 Specification. RFC 3060 (Proposed Standard), February 2001. Updated by RFC 3460.

[92] I. Mulder, B. Hulsebosch, G. Lenzini, and M. S. Bargh. Reading the tea-leaves in an intelligent coffee corner: understanding behavior by using sensory data. In *Proceedings of Measuring Behavior 2008 6th International Conference on Methods and Techniques in Behavioral Research*, 2008.

[93] Ricardo Neisse, Patrícia Dockhorn Costa, Maarten Wegdam, and Marten van Sinderen. Context-aware management domains. *First International Workshop on Combining Context with Trust, Security and Privacy (CAT)*, July 2007.

[94] Ricardo Neisse, Patrícia Dockhorn Costa, Maarten Wegdam, and Marten van Sinderen. An information model and architecture for context-aware management domains. *IEEE Workshop on Policies for Distributed Systems and Networks (POLICY 2008), Palisades, NY, USA*, June 2008.

[95] Ricardo Neisse, Dominik Holling, and Alexander Pretschner. Implementing trust in cloud infrastructures. *Proc. 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*, 2011.

[96] Ricardo Neisse, Alexander Pretschner, and Valentina Di Giacomo. A trustworthy usage control enforcement framework. *Proc. 6th Intl. Conf. on Availability, Reliability and Security (ARES)*, 2011.

[97] Ricardo Neisse, Maarten Wegdam, and Marten van Sinderen. Context-aware trust domains. In *Proceedings of the First European Conference on Smart Sensing and Context, EuroSSC 2006, Enschede, The Netherlands*, volume 4272 of *Lecture Notes in Computer Science*, pages 234–237. Springer Verlag, 2006.

[98] Ricardo Neisse, Maarten Wegdam, and Marten van Sinderen. A distributed context-aware trust management architecture. *Adjunct Proceedings of the 4th International Conference on Pervasive Computing (PERCOM), Dublin, Ireland*, May 2006.

[99] Ricardo Neisse, Maarten Wegdam, and Marten van Sinderen. Trustworthiness and quality of context information. *Proceedings of the 2008 International Symposium on Trusted Computing (TrustCom)*, Nov 2008.

[100] Ricardo Neisse, Maarten Wegdam, Marten van Sinderen, and Gabriele Lenzini. Trust management model and architecture for context-aware service platforms. In *Proceedings of the 2nd International Symposium on Information Security (IS07), November 26-27, Vilamoura, Portugal*, pages 1803–1820, 2007.

[101] Mogens Nielsen and Karl Krukow. Towards a formal notion of trust. In *PPDP '03: Proceedings of the 5th ACM SIGPLAN International Conference on Principles and Practice of Declarative Programming*, pages 4–7, New York, NY, USA, 2003. ACM.

[102] Openview. A primer on policy-based network management. version 1.0. openview network. Available at: `http://www.openview.hp.com/Uploads/primer_on_policy-based_network_mgmt.pdf`, 1999.

[103] Jaehong Park and Ravi Sandhu. The uconabc usage control model. *ACM Trans. Inf. Syst. Secur.*, 7(1):128–174, 2004.

[104] Alexander P. Pons. Biometric marketing: targeting the online consumer. *Commun. ACM*, 49(8):60–66, 2006.

[105] A. Pretschner, M. Hilty, D. Basin, C. Schaefer, and T. Walter. Mechanisms for usage control. In *ASIACCS '08: Proceedings of the 2008 ACM symposium on Information, computer and communications security*, pages 240–244, New York, NY, USA, 2008. ACM.

[106] Liberty Alliance Project. Liberty identity federation framework architecture. Available at: `http://www.projectliberty.org`, 2008.

[107] Karl Quinn, David Lewis, Declan O'Sullivan, and Vincent P. Wade. Trust meta-policies for flexible and dynamic policy based trust management. In *POLICY '06: Proceedings of the Seventh IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY'06)*, pages 145–148, Washington, DC, USA, 2006. IEEE Computer Society.

[108] Daniele Riboni, Linda Pareschi, and Claudio Bettini. Privacy in georeferenced context-aware services: A survey. In *CEUR Workshop Proceedings*, volume 397. CEUR-WS.org, 2008.

[109] RRD. Roessingh research and development. Available at: `http://www.rrd.nl/www/indexa.html`, 2009.

[110] Giovanni Russello, Changyu Dong, and Naranker Dulay. Authorisation and conflict resolution for hierarchical domains. In *POLICY '07: Proceedings of the Eighth IEEE International Workshop on Policies for Distributed Systems and Networks*, pages 201–210, Washington, DC, USA, 2007. IEEE Computer Society.

[111] R.S. Sandhu. Lattice-based access control models. *IEEE Computer*, 1993.

[112] R.S. Sandhu, E.J. Coyne, H.L. Feinstein, and C.E. Youman. Role-based access control models. *IEEE Computer*, 1996.

[113] K. Sheikh, M. Wegdam, and M. J. van Sinderen. Quality-of-context and its use for protecting privacy in context aware systems. *Journal of Software (JSW)*, 3(3):83–93, Mar. 2008.

[114] Santtu Toivonen and Grit Denker. The impact of context on the trustworthiness of communication: An ontological approach. In *ISWC*04 Workshop on Trust, Security, and Reputation on the Semantic Web, Hiroshima, Japan*, volume 127 of *CEUR Workshop Proceedings*, 2004.

[115] Santtu Toivonen, Gabriele Lenzini, and Ilkka Uusitalo. Context-aware trust evaluation functions for dynamic reconfigurable systems. In *Proceedings of the WWW'06 Workshop on Models of Trust for the Web (MTW'06), Edinburgh, Scotland*, 2006.

[116] K. Twidle, E. Lupu, N. Dulay, and M. Sloman. Ponder2 - a policy environment for autonomous pervasive systems. In *IEEE Workshop on Policies for Distributed Systems and Network (POLICY 2008)*, pages 245–246, June 2008.

[117] Kevin Twidle. Ponder2 wiki. Available at: `http://www.ponder2.net`, 2011.

[118] H. van Kranenburg, M.S. Bargh, S. Iacob, and A. Peddemors. A context management framework for supporting context-aware distributed applications. *Communications Magazine, IEEE*, 44(8):67–74, Aug. 2006.

[119] M. J. van Sinderen, A. T. van Halteren, M. Wegdam, H. B. Meeuwissen, and E. H. Eertink. Supporting context-aware mobile applications: an infrastructure approach. *Communications Magazine, IEEE*, 44(9):96–104, Sept. 2006.

[120] R. V. van Wielink. Complexity and Granularity of User Privacy Preferences in Context-Aware Systems, November 2007. Master Thesis, University of Twente, Enschede.

[121] Steve Vinoski. Service discovery 101. *IEEE Internet Computing*, 7:69–71, January 2003.

[122] R. Wieringa. Writing a report about design research. Available at: `http://www.cs.utwente.nl/roelw/`, 2007.

[123] Wikipedia. Statistical survey - advantages and disadvantages of surveys. Available at: `http://en.wikipedia.org/wiki/Statistical_survey`, 2009.

[124] Working Package 3 (WP3). D3.9: Study on the impact of trusted computing on identity and identity management. Available at: `http://www.fidis.net/resources/deliverables/hightechid/`, 2008. FIDIS - Future of Identity in the Information Society.

[125] M. Zuidweg, J. Goncalves Filho, and M.J. van Sinderen. Using p3p in a web services-based context-aware application platform, 2003. 9th EUNICE Open European Summer School and IFIP Workshop on Next Generation Networks (EUNICE 2003).

# SIKS Dissertatiereeks

====
1998
====

1998-1 Johan van den Akker (CWI)
DEGAS - An Active, Temporal Database of Autonomous Objects

1998-2 Floris Wiesman (UM)
Information Retrieval by Graphically Browsing Meta-Information

1998-3 Ans Steuten (TUD)
A Contribution to the Linguistic Analysis of Business Conversations within the Language/Action Perspective

1998-4 Dennis Breuker (UM)
Memory versus Search in Games

1998-5 E.W.Oskamp (RUL)
Computerondersteuning bij Straftoemeting

====
1999
====

1999-1 Mark Sloof (VU)
Physiology of Quality Change Modelling;
Automated modelling of Quality Change of Agricultural Products

1999-2 Rob Potharst (EUR)
Classification using decision trees and neural nets

1999-3 Don Beal (UM)
The Nature of Minimax Search

1999-4 Jacques Penders (UM)
The practical Art of Moving Physical Objects

1999-5 Aldo de Moor (KUB)
Empowering Communities: A Method for the Legitimate User-Driven Specification of Network Information Systems

1999-6 Niek J.E. Wijngaards (VU)
Re-design of compositional systems

1999-7 David Spelt (UT)
Verification support for object database design

1999-8 Jacques H.J. Lenting (UM)
Informed Gambling: Conception and Analysis of a Multi-Agent Mechanism for Discrete Reallocation.

====
2000
====

2000-1 Frank Niessink (VU)
Perspectives on Improving Software Maintenance

2000-2 Koen Holtman (TUE)
Prototyping of CMS Storage Management

2000-3 Carolien M.T. Metselaar (UVA)
Sociaal-organisatorische gevolgen van kennistechnologie;
een procesbenadering en actorperspectief.

2000-4 Geert de Haan (VU)
ETAG, A Formal Model of Competence Knowledge for User Interface Design

2000-5 Ruud van der Pol (UM)
Knowledge-based Query Formulation in Information Retrieval.

2000-6 Rogier van Eijk (UU)
Programming Languages for Agent Communication

2000-7 Niels Peek (UU)
Decision-theoretic Planning of Clinical Patient Management

2000-8 Veerle Coup (EUR)
Sensitivity Analyis of Decision-Theoretic Networks

2000-9 Florian Waas (CWI)
Principles of Probabilistic Query Optimization

2000-10 Niels Nes (CWI)
Image Database Management System Design Considerations,
Algorithms and Architecture

2000-11 Jonas Karlsson (CWI)
Scalable Distributed Data Structures for Database Management

====
2001
====

2001-1 Silja Renooij (UU)
Qualitative Approaches to Quantifying Probabilistic Networks

2001-2 Koen Hindriks (UU)
Agent Programming Languages: Programming with Mental Models

2001-3 Maarten van Someren (UvA)
Learning as problem solving

2001-4 Evgueni Smirnov (UM)
Conjunctive and Disjunctive Version Spaces with
Instance-Based Boundary Sets

2001-5 Jacco van Ossenbruggen (VU)
Processing Structured Hypermedia: A Matter of Style

2001-6 Martijn van Welie (VU)
Task-based User Interface Design

2001-7 Bastiaan Schonhage (VU)
Diva: Architectural Perspectives on Information Visualization

2001-8 Pascal van Eck (VU)
A Compositional Semantic Structure for Multi-Agent Systems Dynamics.

2001-9 Pieter Jan 't Hoen (RUL)
Towards Distributed Development of Large Object-Oriented Models,
Views of Packages as Classes

2001-10 Maarten Sierhuis (UvA)
Modeling and Simulating Work Practice
BRAHMS: a multiagent modeling and simulation language
for work practice analysis and design

2001-11 Tom M. van Engers (VUA)
Knowledge Management:
The Role of Mental Models in Business Systems Design

====
2002
====

2002-01 Nico Lassing (VU)
Architecture-Level Modifiability Analysis

2002-02 Roelof van Zwol (UT)
Modelling and searching web-based document collections

2002-03 Henk Ernst Blok (UT)
Database Optimization Aspects for Information Retrieval

2002-04 Juan Roberto Castelo Valdueza (UU)
The Discrete Acyclic Digraph Markov Model in Data Mining

2002-05 Radu Serban (VU)
The Private Cyberspace Modeling Electronic Environments
inhabited by Privacy-concerned Agents

2002-06 Laurens Mommers (UL)
Applied legal epistemology;
Building a knowledge-based ontology of the legal domain

2002-07 Peter Boncz (CWI)
Monet: A Next-Generation DBMS Kernel For Query-Intensive Applications

2002-08 Jaap Gordijn (VU)
Value Based Requirements Engineering: Exploring Innovative
E-Commerce Ideas

2002-09 Willem-Jan van den Heuvel(KUB)
Integrating Modern Business Applications with Objectified Legacy Systems

2002-10 Brian Sheppard (UM)
Towards Perfect Play of Scrabble

2002-11 Wouter C.A. Wijngaards (VU)
Agent Based Modelling of Dynamics: Biological and Organisational Applications

2002-12 Albrecht Schmidt (Uva)
Processing XML in Database Systems

2002-13 Hongjing Wu (TUE)
A Reference Architecture for Adaptive Hypermedia Applications

2002-14 Wieke de Vries (UU)
Agent Interaction: Abstract Approaches to Modelling, Programming and
Verifying Multi-Agent Systems

2002-15 Rik Eshuis (UT)
Semantics and Verification of UML Activity Diagrams for Workflow Modelling

2002-16 Pieter van Langen (VU)
The Anatomy of Design: Foundations, Models and Applications

2002-17 Stefan Manegold (UVA)
Understanding, Modeling, and Improving Main-Memory Database Performance

====
2003
====

2003-01 Heiner Stuckenschmidt (VU)
Ontology-Based Information Sharing in Weakly Structured Environments

2003-02 Jan Broersen (VU)
Modal Action Logics for Reasoning About Reactive Systems

2003-03 Martijn Schuemie (TUD)
Human-Computer Interaction and Presence in Virtual Reality Exposure Therapy

2003-04 Milan Petkovic (UT)
Content-Based Video Retrieval Supported by Database Technology

2003-05 Jos Lehmann (UVA)
Causation in Artificial Intelligence and Law - A modelling approach

2003-06 Boris van Schooten (UT)
Development and specification of virtual environments

2003-07 Machiel Jansen (UvA)
Formal Explorations of Knowledge Intensive Tasks

2003-08 Yongping Ran (UM)
Repair Based Scheduling

2003-09 Rens Kortmann (UM)
The resolution of visually guided behaviour

2003-10 Andreas Lincke (UvT)
Electronic Business Negotiation: Some experimental studies on the interaction between medium, innovation context and culture

2003-11 Simon Keizer (UT)
Reasoning under Uncertainty in Natural Language Dialogue using Bayesian Networks

2003-12 Roeland Ordelman (UT)
Dutch speech recognition in multimedia information retrieval

2003-13 Jeroen Donkers (UM)
Nosce Hostem - Searching with Opponent Models

2003-14 Stijn Hoppenbrouwers (KUN)
Freezing Language: Conceptualisation Processes across ICT-Supported Organisations

2003-15 Mathijs de Weerdt (TUD)
Plan Merging in Multi-Agent Systems

2003-16 Menzo Windhouwer (CWI)
Feature Grammar Systems - Incremental Maintenance of Indexes to Digital Media Warehouses

2003-17 David Jansen (UT)
Extensions of Statecharts with Probability, Time, and Stochastic Timing

2003-18 Levente Kocsis (UM)
Learning Search Decisions

====
2004
====

2004-01 Virginia Dignum (UU)
A Model for Organizational Interaction: Based on Agents, Founded in Logic

2004-02 Lai Xu (UvT)
Monitoring Multi-party Contracts for E-business

2004-03 Perry Groot (VU)
A Theoretical and Empirical Analysis of Approximation in Symbolic Problem Solving

2004-04 Chris van Aart (UVA)
Organizational Principles for Multi-Agent Architectures

2004-05 Viara Popova (EUR)
Knowledge discovery and monotonicity

2004-06 Bart-Jan Hommes (TUD)
The Evaluation of Business Process Modeling Techniques

2004-07 Elise Boltjes (UM)
Voorbeeldig onderwijs; voorbeeldgestuurd onderwijs, een opstap naar abstract denken, vooral voor meisjes

2004-08 Joop Verbeek(UM)
Politie en de Nieuwe Internationale Informatiemarkt, Grensregionale politiële gegevensuitwisseling en digitale expertise

2004-09 Martin Caminada (VU)
For the Sake of the Argument; explorations into argument-based reasoning

2004-10 Suzanne Kabel (UVA)
Knowledge-rich indexing of learning-objects

2004-11 Michel Klein (VU)
Change Management for Distributed Ontologies

2004-12 The Duy Bui (UT)
Creating emotions and facial expressions for embodied agents

2004-13 Wojciech Jamroga (UT)
Using Multiple Models of Reality: On Agents who Know how to Play

2004-14 Paul Harrenstein (UU)
Logic in Conflict. Logical Explorations in Strategic Equilibrium

2004-15 Arno Knobbe (UU)
Multi-Relational Data Mining

2004-16 Federico Divina (VU)
Hybrid Genetic Relational Search for Inductive Learning

2004-17 Mark Winands (UM)
Informed Search in Complex Games

2004-18 Vania Bessa Machado (UvA)
Supporting the Construction of Qualitative Knowledge Models

2004-19 Thijs Westerveld (UT)
Using generative probabilistic models for multimedia retrieval

2004-20 Madelon Evers (Nyenrode)
Learning from Design: facilitating multidisciplinary design teams

====
2005
====

2005-01 Floor Verdenius (UVA)
Methodological Aspects of Designing Induction-Based Applications

2005-02 Erik van der Werf (UM))
AI techniques for the game of Go

2005-03 Franc Grootjen (RUN)
A Pragmatic Approach to the Conceptualisation of Language

2005-04 Nirvana Meratnia (UT)
Towards Database Support for Moving Object data

2005-05 Gabriel Infante-Lopez (UVA)
Two-Level Probabilistic Grammars for Natural Language Parsing

2005-06 Pieter Spronck (UM)
Adaptive Game AI

2005-07 Flavius Frasincar (TUE)
Hypermedia Presentation Generation for Semantic Web Information Systems

2005-08 Richard Vdovjak (TUE)
A Model-driven Approach for Building Distributed Ontology-based Web Applications

2005-09 Jeen Broekstra (VU)
Storage, Querying and Inferencing for Semantic Web Languages

2005-10 Anders Bouwer (UVA)
Explaining Behaviour: Using Qualitative Simulation in Interactive Learning Environments

2005-11 Elth Ogston (VU)
Agent Based Matchmaking and Clustering - A Decentralized Approach to Search

2005-12 Csaba Boer (EUR)
Distributed Simulation in Industry

2005-13 Fred Hamburg (UL)
Een Computermodel voor het Ondersteunen van Euthanasiebeslissingen

2005-14 Borys Omelayenko (VU)
Web-Service configuration on the Semantic Web; Exploring how semantics meets pragmatics

2005-15 Tibor Bosse (VU)
Analysis of the Dynamics of Cognitive Processes

2005-16 Joris Graaumans (UU)
Usability of XML Query Languages

2005-17 Boris Shishkov (TUD)
Software Specification Based on Re-usable Business Components

2005-18 Danielle Sent (UU)
Test-selection strategies for probabilistic networks

2005-19 Michel van Dartel (UM)
Situated Representation

2005-20 Cristina Coteanu (UL)
Cyber Consumer Law, State of the Art and Perspectives

2005-21 Wijnand Derks (UT)
Improving Concurrency and Recovery in Database Systems by
Exploiting Application Semantics

====
2006
====

2006-01 Samuil Angelov (TUE)
Foundations of B2B Electronic Contracting

2006-02 Cristina Chisalita (VU)
Contextual issues in the design and use of information technology in organizations

2006-03 Noor Christoph (UVA)
The role of metacognitive skills in learning to solve problems

2006-04 Marta Sabou (VU)
Building Web Service Ontologies

2006-05 Cees Pierik (UU)
Validation Techniques for Object-Oriented Proof Outlines

2006-06 Ziv Baida (VU)
Software-aided Service Bundling - Intelligent Methods & Tools
for Graphical Service Modeling

2006-07 Marko Smiljanic (UT)
XML schema matching – balancing efficiency and effectiveness by means of clustering

2006-08 Eelco Herder (UT)
Forward, Back and Home Again - Analyzing User Behavior on the Web

2006-09 Mohamed Wahdan (UM)
Automatic Formulation of the Auditor's Opinion

2006-10 Ronny Siebes (VU)
Semantic Routing in Peer-to-Peer Systems

2006-11 Joeri van Ruth (UT)
Flattening Queries over Nested Data Types

2006-12 Bert Bongers (VU)
Interactivation - Towards an e-cology of people, our technological environment, and the arts

2006-13 Henk-Jan Lebbink (UU)
Dialogue and Decision Games for Information Exchanging Agents

2006-14 Johan Hoorn (VU)
Software Requirements: Update, Upgrade, Redesign - towards a Theory of Requirements Change

2006-15 Rainer Malik (UU)
CONAN: Text Mining in the Biomedical Domain

2006-16 Carsten Riggelsen (UU)
Approximation Methods for Efficient Learning of Bayesian Networks

2006-17 Stacey Nagata (UU)
User Assistance for Multitasking with Interruptions on a Mobile Device

2006-18 Valentin Zhizhkun (UVA)
Graph transformation for Natural Language Processing

2006-19 Birna van Riemsdijk (UU)
Cognitive Agent Programming: A Semantic Approach

2006-20 Marina Velikova (UvT)
Monotone models for prediction in data mining

2006-21 Bas van Gils (RUN)
Aptness on the Web

2006-22 Paul de Vrieze (RUN)
Fundaments of Adaptive Personalisation

2006-23 Ion Juvina (UU)
Development of Cognitive Model for Navigating on the Web

2006-24 Laura Hollink (VU)
Semantic Annotation for Retrieval of Visual Resources

2006-25 Madalina Drugan (UU)
Conditional log-likelihood MDL and Evolutionary MCMC

2006-26 Vojkan Mihajloviç (UT)
Score Region Algebra: A Flexible Framework for Structured Information Retrieval

2006-27 Stefano Bocconi (CWI)
Vox Populi: generating video documentaries from semantically annotated media repositories

2006-28 Borkur Sigurbjornsson (UVA)
Focused Information Access using XML Element Retrieval

====
2007
====

2007-01 Kees Leune (UvT)
Access Control and Service-Oriented Architectures

2007-02 Wouter Teepe (RUG)
Reconciling Information Exchange and Confidentiality: A Formal Approach

2007-03 Peter Mika (VU)
Social Networks and the Semantic Web

2007-04 Jurriaan van Diggelen (UU)
Achieving Semantic Interoperability in Multi-agent Systems: a dialogue-based approach

2007-05 Bart Schermer (UL)
Software Agents, Surveillance, and the Right to Privacy: a Legislative Framework for Agent-enabled Surveillance

2007-06 Gilad Mishne (UVA)
Applied Text Analytics for Blogs

2007-07 Natasa Jovanovic' (UT)
To Whom It May Concern - Addressee Identification in Face-to-Face Meetings

2007-08 Mark Hoogendoorn (VU)
Modeling of Change in Multi-Agent Organizations

2007-09 David Mobach (VU)
Agent-Based Mediated Service Negotiation

2007-10 Huib Aldewereld (UU)
Autonomy vs. Conformity: an Institutional Perspective on Norms and Protocols

2007-11 Natalia Stash (TUE)
Incorporating Cognitive/Learning Styles in a General-Purpose Adaptive Hypermedia System

2007-12 Marcel van Gerven (RUN)
Bayesian Networks for Clinical Decision Support: A Rational Approach to Dynamic Decision-Making under Uncertainty

2007-13 Rutger Rienks (UT)
Meetings in Smart Environments; Implications of Progressing Technology

2007-14 Niek Bergboer (UM)
Context-Based Image Analysis

2007-15 Joyca Lacroix (UM)
NIM: a Situated Computational Memory Model

2007-16 Davide Grossi (UU)
Designing Invisible Handcuffs. Formal investigations in Institutions and Organizations for Multi-agent Systems

2007-17 Theodore Charitos (UU)
Reasoning with Dynamic Networks in Practice

2007-18 Bart Orriens (UvT)
On the development an management of adaptive business collaborations

2007-19 David Levy (UM)
Intimate relationships with artificial partners

2007-20 Slinger Jansen (UU)
Customer Configuration Updating in a Software Supply Network

2007-21 Karianne Vermaas (UU)
Fast diffusion and broadening use: A research on residential adoption and usage of broadband internet in the Netherlands between 2001 and 2005

2007-22 Zlatko Zlatev (UT)
Goal-oriented design of value and process models from patterns

2007-23 Peter Barna (TUE)
Specification of Application Logic in Web Information Systems

2007-24 Georgina Ramírez Camps (CWI)
Structural Features in XML Retrieval

2007-25 Joost Schalken (VU)
Empirical Investigations in Software Process Improvement

====
2008
====

2008-01 Katalin Boer-Sorbán (EUR)
Agent-Based Simulation of Financial Markets: A modular,continuous-time approach

2008-02 Alexei Sharpanskykh (VU)
On Computer-Aided Methods for Modeling and Analysis of Organizations

2008-03 Vera Hollink (UVA)
Optimizing hierarchical menus: a usage-based approach

2008-04 Ander de Keijzer (UT)
Management of Uncertain Data - towards unattended integration

2008-05 Bela Mutschler (UT)
Modeling and simulating causal dependencies on process-aware information systems from a cost perspective

2008-06 Arjen Hommersom (RUN)
On the Application of Formal Methods to Clinical Guidelines, an Artificial Intelligence Perspective

2008-07 Peter van Rosmalen (OU)
Supporting the tutor in the design and support of adaptive e-learning

2008-08 Janneke Bolt (UU)
Bayesian Networks: Aspects of Approximate Inference

2008-09 Christof van Nimwegen (UU)
The paradox of the guided user: assistance can be counter-effective

2008-10 Wauter Bosma (UT)
Discourse oriented summarization

2008-11 Vera Kartseva (VU)
Designing Controls for Network Organizations: A Value-Based Approach

2008-12 Jozsef Farkas (RUN)
A Semiotically Oriented Cognitive Model of Knowledge Representation

2008-13 Caterina Carraciolo (UVA)
Topic Driven Access to Scientific Handbooks

2008-14 Arthur van Bunningen (UT)
Context-Aware Querying; Better Answers with Less Effort

2008-15 Martijn van Otterlo (UT)
The Logic of Adaptive Behavior: Knowledge Representation and Algorithms for the Markov Decision Process Framework in First-Order Domains.

2008-16 Henriette van Vugt (VU)
Embodied agents from a user's perspective

2008-17 Martin Op 't Land (TUD)
Applying Architecture and Ontology to the Splitting and Allying of Enterprises

2008-18 Guido de Croon (UM)
Adaptive Active Vision

2008-19 Henning Rode (UT)
From Document to Entity Retrieval: Improving Precision and Performance of Focused Text Search

2008-20 Rex Arendsen (UVA)
Geen bericht, goed bericht. Een onderzoek naar de effecten van de introductie van elektronisch berichtenverkeer met de overheid op de administratieve lasten van bedrijven

2008-21 Krisztian Balog (UVA)
People Search in the Enterprise

2008-22 Henk Koning (UU)
Communication of IT-Architecture

2008-23 Stefan Visscher (UU)
Bayesian network models for the management of ventilator-associated pneumonia

2008-24 Zharko Aleksovski (VU)
Using background knowledge in ontology matching

2008-25 Geert Jonker (UU)
Efficient and Equitable Exchange in Air Traffic Management Plan Repair using Spender-signed Currency

2008-26 Marijn Huijbregts (UT)
Segmentation, Diarization and Speech Transcription: Surprise Data Unraveled

2008-27 Hubert Vogten (OU)
Design and Implementation Strategies for IMS Learning Design

2008-28 Ildiko Flesch (RUN)
On the Use of Independence Relations in Bayesian Networks

2008-29 Dennis Reidsma (UT)
Annotations and Subjective Machines - Of Annotators, Embodied Agents, Users, and Other Humans

2008-30 Wouter van Atteveldt (VU)
Semantic Network Analysis: Techniques for Extracting, Representing and Querying Media Content

2008-31 Loes Braun (UM)
Pro-Active Medical Information Retrieval

2008-32 Trung H. Bui (UT)
Toward Affective Dialogue Management using Partially Observable Markov Decision Processes

2008-33 Frank Terpstra (UVA)
Scientific Workflow Design; theoretical and practical issues

2008-34 Jeroen de Knijf (UU)
Studies in Frequent Tree Mining

2008-35 Ben Torben Nielsen (UvT)
Dendritic morphologies: function shapes structure

====
2009
====

2009-01 Rasa Jurgelenaite (RUN)
Symmetric Causal Independence Models

2009-02 Willem Robert van Hage (VU)
Evaluating Ontology-Alignment Techniques

2009-03 Hans Stol (UvT)
A Framework for Evidence-based Policy Making Using IT

2009-04 Josephine Nabukenya (RUN)
Improving the Quality of Organisational Policy Making using Collaboration Engineering

2009-05 Sietse Overbeek (RUN)
Bridging Supply and Demand for Knowledge Intensive Tasks - Based on Knowledge, Cognition, and Quality

2009-06 Muhammad Subianto (UU)
Understanding Classification

2009-07 Ronald Poppe (UT)
Discriminative Vision-Based Recovery and Recognition of Human Motion

2009-08 Volker Nannen (VU)
Evolutionary Agent-Based Policy Analysis in Dynamic Environments

2009-09 Benjamin Kanagwa (RUN)
Design, Discovery and Construction of Service-oriented Systems

2009-10 Jan Wielemaker (UVA)
Logic programming for knowledge-intensive interactive applications

2009-11 Alexander Boer (UVA)
Legal Theory, Sources of Law & the Semantic Web

2009-12 Peter Massuthe (TUE, Humboldt-Universitaet zu Berlin)
Operating Guidelines for Services

2009-13 Steven de Jong (UM)
Fairness in Multi-Agent Systems

2009-14 Maksym Korotkiy (VU)
From ontology-enabled services to service-enabled ontologies (making ontologies work in e-science with ONTO-SOA)

2009-15 Rinke Hoekstra (UVA)
Ontology Representation - Design Patterns and Ontologies that Make Sense

2009-16 Fritz Reul (UvT)
New Architectures in Computer Chess

2009-17 Laurens van der Maaten (UvT)
Feature Extraction from Visual Data

2009-18 Fabian Groffen (CWI)
Armada, An Evolving Database System

2009-19 Valentin Robu (CWI)
Modeling Preferences, Strategic Reasoning and Collaboration in Agent-Mediated Electronic Markets

2009-20 Bob van der Vecht (UU)
Adjustable Autonomy: Controling Influences on Decision Making

2009-21 Stijn Vanderlooy (UM)
Ranking and Reliable Classification

2009-22 Pavel Serdyukov (UT)
Search For Expertise: Going beyond direct evidence

2009-23 Peter Hofgesang (VU)
Modelling Web Usage in a Changing Environment

2009-24 Annerieke Heuvelink (VUA)
Cognitive Models for Training Simulations

2009-25 Alex van Ballegooij (CWI)
"RAM: Array Database Management through Relational Mapping"

2009-26 Fernando Koch (UU)
An Agent-Based Model for the Development of Intelligent Mobile Services

2009-27 Christian Glahn (OU)
Contextual Support of social Engagement and Reflection on the Web

2009-28 Sander Evers (UT)
Sensor Data Management with Probabilistic Models

2009-29 Stanislav Pokraev (UT)
Model-Driven Semantic Integration of Service-Oriented Applications

2009-30 Marcin Zukowski (CWI)
Balancing vectorized query execution with bandwidth-optimized storage

2009-31 Sofiya Katrenko (UVA)
A Closer Look at Learning Relations from Text

2009-32 Rik Farenhorst (VU) and Remco de Boer (VU)
Architectural Knowledge Management: Supporting Architects and Auditors

2009-33 Khiet Truong (UT)
How Does Real Affect Affect Affect Recognition In Speech?

2009-34 Inge van de Weerd (UU)
Advancing in Software Product Management: An Incremental Method Engineering Approach

2009-35 Wouter Koelewijn (UL)
Privacy en Politiegegevens; Over geautomatiseerde normatieve informatie-uitwisseling

2009-36 Marco Kalz (OUN)
Placement Support for Learners in Learning Networks

2009-37 Hendrik Drachsler (OUN)
Navigation Support for Learners in Informal Learning Networks

2009-38 Riina Vuorikari (OU)
Tags and self-organisation: a metadata ecology for learning resources in a multilingual context

2009-39 Christian Stahl (TUE, Humboldt-Universitaet zu Berlin)
Service Substitution – A Behavioral Approach Based on Petri Nets

2009-40 Stephan Raaijmakers (UvT)
Multinomial Language Learning: Investigations into the Geometry of Language

2009-41 Igor Berezhnyy (UvT)
Digital Analysis of Paintings

2009-42 Toine Bogers
Recommender Systems for Social Bookmarking

2009-43 Virginia Nunes Leal Franqueira (UT)
Finding Multi-step Attacks in Computer Networks using Heuristic Search and Mobile Ambients

2009-44 Roberto Santana Tapia (UT)
Assessing Business-IT Alignment in Networked Organizations

2009-45 Jilles Vreeken (UU)
Making Pattern Mining Useful

2009-46 Loredana Afanasiev (UvA)
Querying XML: Benchmarks and Recursion

====
2010
====

2010-01 Matthijs van Leeuwen (UU)
Patterns that Matter

2010-02 Ingo Wassink (UT)
Work flows in Life Science

2010-03 Joost Geurts (CWI)
A Document Engineering Model and Processing Framework for Multimedia documents

2010-04 Olga Kulyk (UT)
Do You Know What I Know? Situational Awareness of Co-located Teams in Multidisplay Environments

2010-05 Claudia Hauff (UT)
Predicting the Effectiveness of Queries and Retrieval Systems

2010-06 Sander Bakkes (UvT)
Rapid Adaptation of Video Game AI

2010-07 Wim Fikkert (UT)
Gesture interaction at a Distance

2010-08 Krzysztof Siewicz (UL)
Towards an Improved Regulatory Framework of Free Software. Protecting user freedoms in a world of software communities and eGovernments

2010-09 Hugo Kielman (UL)
A Politiele gegevensverwerking en Privacy, Naar een effectieve waarborging

2010-10 Rebecca Ong (UL)
Mobile Communication and Protection of Children 2010-11 Adriaan Ter Mors (TUD)
The world according to MARP: Multi-Agent Route Planning

2010-12 Susan van den Braak (UU)
Sensemaking software for crime analysis

2010-13 Gianluigi Folino (RUN)
High Performance Data Mining using Bio-inspired techniques

2010-14 Sander van Splunter (VU)
Automated Web Service Reconfiguration

2010-15 Lianne Bodenstaff (UT)
Managing Dependency Relations in Inter-Organizational Models

2010-16 Sicco Verwer (TUD)
Efficient Identification of Timed Automata, theory and practice

2010-17 Spyros Kotoulas (VU)
Scalable Discovery of Networked Resources: Algorithms, Infrastructure, Applications

2010-18 Charlotte Gerritsen (VU)
Caught in the Act: Investigating Crime by Agent-Based Simulation

2010-19 Henriette Cramer (UvA)
People's Responses to Autonomous and Adaptive Systems

2010-20 Ivo Swartjes (UT)
Whose Story Is It Anyway? How Improv Informs Agency and Authorship of Emergent Narrative

2010-21 Harold van Heerde (UT)
Privacy-aware data management by means of data degradation

2010-22 Michiel Hildebrand (CWI)
End-user Support for Access to Heterogeneous Linked Data

2010-23 Bas Steunebrink (UU)
The Logical Structure of Emotions

2010-24 Dmytro Tykhonov
Designing Generic and Efficient Negotiation Strategies

2010-25 Zulfiqar Ali Memon (VU)
Modelling Human-Awareness for Ambient Agents: A Human Mindreading Perspective

2010-26 Ying Zhang (CWI)
XRPC: Efficient Distributed Query Processing on Heterogeneous XQuery Engines

2010-27 Marten Voulon (UL)
Automatisch contracteren

2010-28 Arne Koopman (UU)
Characteristic Relational Patterns

2010-29 Stratos Idreos(CWI)
Database Cracking: Towards Auto-tuning Database Kernels

2010-30 Marieke van Erp (UvT)
Accessing Natural History - Discoveries in data cleaning, structuring, and retrieval

2010-31 Victor de Boer (UVA)
Ontology Enrichment from Heterogeneous Sources on the Web

2010-32 Marcel Hiel (UvT)
An Adaptive Service Oriented Architecture: Automatically solving Interoperability Problems

2010-33 Robin Aly (UT)
Modeling Representation Uncertainty in Concept-Based Multimedia Retrieval

2010-34 Teduh Dirgahayu (UT)
Interaction Design in Service Compositions

2010-35 Dolf Trieschnigg (UT)
Proof of Concept: Concept-based Biomedical Information Retrieval

2010-36 Jose Janssen (OU)
Paving the Way for Lifelong Learning; Facilitating competence development through a learning path specification

2010-37 Niels Lohmann (TUE)
Correctness of services and their composition

2010-38 Dirk Fahland (TUE)
From Scenarios to components

2010-39 Ghazanfar Farooq Siddiqui (VU)
Integrative modeling of emotions in virtual agents

2010-40 Mark van Assem (VU)
Converting and Integrating Vocabularies for the Semantic Web

2010-41 Guillaume Chaslot (UM)
Monte-Carlo Tree Search

2010-42 Sybren de Kinderen (VU)
Needs-driven service bundling in a multi-supplier setting - the computational e3-service approach

2010-43 Peter van Kranenburg (UU)
A Computational Approach to Content-Based Retrieval of Folk Song Melodies

2010-44 Pieter Bellekens (TUE)
An Approach towards Context-sensitive and User-adapted Access to Heterogeneous Data Sources, Illustrated in the Television Domain

2010-45 Vasilios Andrikopoulos (UvT)
A theory and model for the evolution of software services

2010-46 Vincent Pijpers (VU)
e3alignment: Exploring Inter-Organizational Business-ICT Alignment

2010-47 Chen Li (UT)
Mining Process Model Variants: Challenges, Techniques, Examples

2010-48 Milan Lovric (EUR)
Behavioral Finance and Agent-Based Artificial Markets

2010-49 Jahn-Takeshi Saito (UM)
Solving difficult game positions

2010-50 Bouke Huurnink (UVA)
Search in Audiovisual Broadcast Archives

2010-51 Alia Khairia Amin (CWI)
Understanding and supporting information seeking tasks in multiple sources

2010-52 Peter-Paul van Maanen (VU)
Adaptive Support for Human-Computer Teams: Exploring the Use of Cognitive Models of Trust and Attention

2010-53 Edgar Meij (UVA)
Combining Concepts and Language Models for Information Access

====
2011
====

2011-01 Botond Cseke (RUN)
Variational Algorithms for Bayesian Inference in Latent Gaussian Models

2011-02 Nick Tinnemeier(UU)
Organizing Agent Organizations. Syntax and Operational Semantics of an Organization-Oriented Programming Language

2011-03 Jan Martijn van der Werf (TUE)
Compositional Design and Verification of Component-Based Information Systems

2011-04 Hado van Hasselt (UU)
Insights in Reinforcement Learning; Formal analysis and empirical evaluation of temporal-difference learning algorithms

2011-05 Base van der Raadt (VU)
Enterprise Architecture Coming of Age - Increasing the Performance of an Emerging Discipline.

2011-06 Yiwen Wang (TUE)
Semantically-Enhanced Recommendations in Cultural Heritage

2011-07 Yujia Cao (UT)
Multimodal Information Presentation for High Load Human Computer Interaction

2011-08 Nieske Vergunst (UU)
BDI-based Generation of Robust Task-Oriented Dialogues

2011-09 Tim de Jong (OU)
Contextualised Mobile Media for Learning

2011-10 Bart Bogaert (UvT)
Cloud Content Contention

2011-11 Dhaval Vyas (UT)
Designing for Awareness: An Experience-focused HCI Perspective

2011-12 Carmen Bratosin (TUE)
Grid Architecture for Distributed Process Mining

2011-13 Xiaoyu Mao (UvT)
Airport under Control. Multiagent Scheduling for Airport Ground Handling

2011-14 Milan Lovric (EUR)
Behavioral Finance and Agent-Based Artificial Markets

2011-15 Marijn Koolen (UvA)
The Meaning of Structure: the Value of Link Evidence for Information Retrieval

2011-16 Maarten Schadd (UM)
Selective Search in Games of Different Complexity

2011-17 Jiyin He (UVA)
Exploring Topic Structure: Coherence, Diversity and Relatedness

2011-18 Mark Ponsen (UM)
Strategic Decision-Making in complex games

2011-19 Ellen Rusman (OU)
The Mind ' s Eye on Personal Profiles

2011-20 Qing Gu (VU)
Guiding service-oriented software engineering - A view-based approach

2011-21 Linda Terlouw (TUD)
Modularization and Specification of Service-Oriented Systems

2011-22 Junte Zhang (UVA)
System Evaluation of Archival Description and Access

2011-23 Wouter Weerkamp (UVA)
Finding People and their Utterances in Social Media

2011-24 Herwin van Welbergen (UT)
Behavior Generation for Interpersonal Coordination with Virtual Humans On Specifying, Scheduling and Realizing Multimodal Virtual Human Behavior

2011-25 Syed Waqar ul Qounain Jaffry (VU)
Analysis and Validation of Models for Trust Dynamics

2011-26 Matthijs Aart Pontier (VU)
Virtual Agents for Human Communication - Emotion Regulation and Involvement-Distance Trade-Offs in Embodied Conversational Agents and Robots

2011-27 Aniel Bhulai (VU)
Dynamic website optimization through autonomous management of design patterns

2011-28 Rianne Kaptein(UVA)
Effective Focused Retrieval by Exploiting Query Context and Document Structure

2011-29 Faisal Kamiran (TUE)
Discrimination-aware Classification

2011-30 Egon van den Broek (UT)
Affective Signal Processing (ASP): Unraveling the mystery of emotions

2011-31 Ludo Waltman (EUR)
Computational and Game-Theoretic Approaches for Modeling Bounded Rationality

2011-32 Nees-Jan van Eck (EUR)
Methodological Advances in Bibliometric Mapping of Science

2011-33 Tom van der Weide (UU)
Arguing to Motivate Decisions

2011-34 Paolo Turrini (UU)
Strategic Reasoning in Interdependence: Logical and Game-theoretical Investigations

2011-35 Maaike Harbers (UU)
Explaining Agent Behavior in Virtual Training

2011-36 Erik van der Spek (UU)
Experiments in serious game design: a cognitive approach

2011-37 Adriana Burlutiu (RUN)
Machine Learning for Pairwise Data, Applications for Preference Learning and Supervised Network Inference

2011-38 Nyree Lemmens (UM)
Bee-inspired Distributed Optimization

2011-39 Joost Westra (UU)
Organizing Adaptation using Agents in Serious Games

2011-40 Viktor Clerc (VU)
Architectural Knowledge Management in Global Software Development

2011-41 Luan Ibraimi (UT)
Cryptographically Enforced Distributed Data Access Control

2011-42 Michal Sindlar (UU)
Explaining Behavior through Mental State Attribution

2011-43 Henk van der Schuur (UU)
Process Improvement through Software Operation Knowledge

2011-44 Boris Reuderink (UT)
Robust Brain-Computer Interfaces

2011-45 Herman Stehouwer (UvT)
Statistical Language Models for Alternative Sequence Selection

2011-46 Beibei Hu (TUD)
Towards Contextualized Information Delivery: A Rule-based Architecture for the Domain of Mobile Police Work

2011-47 Azizi Bin Ab Aziz(VU)
Exploring Computational Models for Intelligent Support of Persons with Depression

2011-48 Mark Ter Maat (UT)
Response Selection and Turn-taking for a Sensitive Artificial Listening Agent

2011-49 Andreea Niculescu (UT)
Conversational interfaces for task-oriented spoken dialogues: design aspects influencing interaction quality

2012-07 Rianne van Lambalgen (VU)
When the Going Gets Tough: Exploring Agent-based Models of Human-like Performance under Demanding Conditions

2012-08 Gerben de Vries (UVA)
Kernel Methods for Vessel Trajectories

# About the author

Ricardo Neisse was born in Carazinho, Rio Grande do Sul, Brazil, on the 9th of March of 1979. He has a master's degree (MSc) in Computer Science from the Federal University of Rio Grande do Sul, Brazil. Since 2005, he has been investigating trust and privacy issues in context-aware service platforms.

From 2005 to 2009, he worked in the ASNA and IS group at the University of Twente, The Netherlands. In this period, he has participated in the AWARENESS research project, and has developed his PhD research, which resulted in this book. He has authored many international publications including conferences and workshops. He has served as a reviewers and technical program committee member for many international conferences and workshops. Since May 2009, he has been working at the Fraunhofer Institute for Experimental Software Engineering in the area of Distributed Usage Control. In his free time, he enjoys indoor rock climbing and is an enthusiast amateur triathlete.

Below is a list of his publications in reverse chronological order:

- Neisse, R.; Pretschner, A.; Di Giacomo, V. A Trustworthy Usage Control Enforcement Framework. Proc. 6th Intl. Conf. on Availability, Reliability and Security (ARES), Aug. 2011;
- Neisse, R.; Holling, D.; Pretschner, A. Implementing Trust in Cloud Infrastructures. In Proc. 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID), May 2011;
- Neisse, R. Trust Management Models for Distributed Usage Control. Dagstuhl Seminar on Distributed Usage Control, Apr. 2010 (invited talk and abstract);
- Neisse, R. Trust Management in Context-Aware and Service-oriented Architectures. European Workshop on Combining Context with Trust, Security, and Privacy (EuroCAT'09), Pisa, Italy, Sep. 2009 (invited talk and extended abstract);

- Neisse, R. Trust Management in Pervasive and Service-oriented Architectures. IEEE Workshop on Security and High Performance Computing Systems, Leipzig, Germany, June 2009 (invited talk and extended abstract);
- Granville, L. Z.; Neisse, R.; Vianna, R. L.; Fioreze, T. On the Management Performance of Networked Environments Using Web Services Technologies. Chapter in the Handbook of Research on Telecommunications Planning and Management for Business, Mar-2009;
- Neisse, R.; Wegdam, M.; van Sinderen, M. Trustworthiness and Quality of Context Information. The 2008 International Symposium on Trusted Computing (TrustCom), Nov-2008;
- Neisse, R.; Wegdam, M.; Dockhorn Costa, P.; van Sinderen, M. An Information Model and Architecture for Context-Aware Management Domains. IEEE Workshop on Policies for Distributed Systems and Networks (POLICY 2008), Palisades, NY, USA, Jun-2008;
- Neisse, R.; Wegdam, M.; van Sinderen, M.; Lenzini, G. Trust Management Model and Architecture for Context-Aware Service Platforms. The 2nd International Symposium on Information Security (IS'07), Vilamoura, Portugal, Nov-2007;
- Neisse, R.; Wegdam, M.; Dockhorn Costa, P.; van Sinderen, M. Context-Aware Management Domains. First International Workshop on Combining Context with Trust, Security, and Privacy, Moncton, Canada, Jul-2007;
- Pessoa, R.; Calvi, C.; Pereira Filho, J.; Farias, C.; Neisse, R. Semantic Context Reasoning Using Ontology Based Models. 13th EUNICE Open European Summer School and IFIP TC6.6 Workshop on Dependable and Adaptable Networks and Services, Enschede, The Netherlands, Jul-2007;
- Neisse, R.; Wegdam, M.; van Sinderen, M. Context-Aware Trust Domains. 1st European Conference on Smart Sensing and Context, Enschede, The Netherlands, Oct-2006;
- Neisse, R. Distributed Context-Aware Trust Management. CTIT Symposium Smart Environments (awarded 3rd prize poster competition)., May-2006;
- Neisse, R.; Wegdam, M.; van Sinderen, M. A Distributed Context-Aware Trust Management Architecture. Adjunct Proceedings of 4th International Conference on Pervasive Computing (PERVASIVE 2006) - Doctoral Colloquium, Dublin, Ireland, May-2006;
- Fioreze, T.; Neisse, R.; Granville, L. Z.; Almeida, M. J. B.; Pras, A. A Policy-Based Hierarchical Approach for Management of Grids and Networks. Proceedings of 2006 IEEE/IFIP Netwok Operations and Management Symposium (NOMS 2006) - Application Session, Vancouver, Canada, Apr-2006;

- Neisse, R.; Granville, L. Z.; Almeida, M. J. B.; Tarouco, L. M. R. Grid-Aware Network Resources Allocation using a Policy-Based Approach. Proceedings of 22nd Brazilian Symposium on Computer Networks (SBRC), Fortaleza, Brasil, May-2005;
- Neisse, R.; Granville, L. Z.; Almeida, M. J. B.; Tarouco, L. M. R. Policies Translation for Integrated Management of Grids and Networks. Special Track on Distributed Systems and Grid Computing of the 20th Annual ACM Symposium on Applied Computing (SAC2005), Santa Fe, USA, Mar-2005;
- Neisse, R.; Granville, L. Z.; Almeida, M. J. B.; Tarouco, L. M. R. Managing Grids Communication Infrastructure through Policy Translations. Proceedings of 2005 IEEE International Workshop on IP Operations and Management (IPOM2004), Beijing, China, Oct-2004;
- Neisse, R.; Granville, L. Z.; Almeida, M. J. B.; Tarouco, L. M. R. On Translating Grid Requirements to Network Configurations through Policy-Based Management. Proceedings of 5th IEEE/ACM International Workshop on Grid Computing (GRID2004), Pittsburgh, USA, Nov-2004;
- Neisse, R.; Pereira, E. D. V.; Granville, L. Z.; Almeida, M. J. B.; Tarouco, L. M. R. Policy-based Management of Grids and Networks Through an Hierarchical Architecture. Proceedings of The First International Workshop on Service Assurance with Partial and Intermittent Resources (SAPIR) in the 11th International Conference on Telecommunications (ICT2004), Fortaleza, Brazil, Aug-2004;
- Neisse, R.; Pereira, E. D. V.; Granville, L. Z.; Almeida, M. J. B.; Tarouco, L. M. R. An Hierarchical Policy-Based Architecture for Integrated Management of Grids and Networks. Proceedings of the IEEE 5th International Workshop on Policies for Distributed Systems and Networks (POLICY2004), pages 103-106, Yorktown Heights, USA, Jul-2004;
- Vianna, R.; Neisse, R.; Granville, L. Z.; Almeida, M. J. B.; Tarouco, L. M. R. A Tool to Create SNMP to Web Services Gateways. Proceedings of 21st Brazilian Symposium on Computer Networks (SBRC), v. 2, pages 907-914, Gramado, Brazil (in Portuguese), May-2004;
- Neisse, R.; Granville, L. Z.; Almeida, M. J. B.; Tarouco, L. M. R. Policy-based Integrated Management of Networks and Grids. Proceedings of 21st Brazilian Symposium on Computer Networks (SBRC), v. 1, pages 137-140, Gramado, Brazil (in Portuguese), May-2004;
- Neisse, R.; Vianna, R.; Granville, L. Z.; Almeida, M. J. B.; Tarouco, L. M. R. Implementation and Bandwidth Consumption Evaluation of SNMP to Web Services Gateways. Proceedings of the IEEE/IFIP Network Operations and Management Symposium (NOMS), pages 715-728, Seoul, Korea, Apr-2004;

- Neisse, R.; Granville, L. Z.; Almeida, M. J. B.; Tarouco, L. M. R. Integrated Policy-based Management of Grids and Networks. Proceedings of II Workshop of Computing Grids and Appplications (WGCA), Petrópolis, Brazil. (invited talk in Portuguese), Feb-2004;
- Neisse, R.; Granville, L. Z.; Almeida, M. J. B.; Tarouco, L. M. R. A Web-based System to Monitor and Analyze Network Management Information in XML. Proceedings of the 3rd IEEE Latin American Network Operations and Management Symposium (LANOMS), Iguassu Falls, Brazil, Sep-2003;
- Neisse, R.; Granville, L. Z.; Almeida, M. J. B.; Tarouco, L. M. R. A Proxy Creation Service for SNMP to XML Translations. Proceedings of the 3rd IEEE Latin American Network Operations and Management Symposium (LANOMS), Iguassu Falls, Brazil, Sep-2003;
- Neisse, R.; Vaguetti, L.; Granville, L. Z.; Almeida, M. J. B.; Tarouco, L. M. R. An Architecture for Integrated Management of Grids. UFRGS Network Management Seminar, Cadernos de Informatica, Porto Alegre, Brazil (in Portuguese), Apr-2003;
- Neisse, R.; Ballve, D. O.; Granville, L. Z.; Almeida, M. J. B.; Tarouco, L. M. R. A Dynamic SNMP to XML Proxy. Proceedings of the 8th IFIP/IEEE International Symposium on Integrated Network Management (IM), Boston: Kluwer Academic Publishers, v. 1, pages 481-484, Colorado Springs, USA, Mar-2003;
- Neisse, R.; Granville, L. Z.; Ballve, D. O.; Almeida, M. J. B.; Tarouco, L. M. R. A Solution to Dynamically create SNMP to XML Proxies. Proceedings of the International Conference on Information Networking (ICOIN), v. 1, pages 543-552, Jeju Island, Korea, Feb-2003;
- Leonhardt, M. D.; Neisse, R.; Granville, L. Z.; Almeida, M. J. B.; Tarouco, L. M. R. MEARA: A Thematic Chatterbot to Use in Educational Environments. XIV Brazilian Symposium on Informatics in Education (SBIE), p. 95-92, Rio de Janeiro, Brazil (in Portuguese), Nov-2003;
- Vaguetti, L.; Neisse, R.; Granville, L. Z.; Almeida, M. J. B.; Tarouco, L. M. R. An Architecture for Integrated Policy-Based Management of QoS and Multicast-Enabled Networks. Proceedings of the 21st Brazilian Symposium on Computer Networks (SBRC), Natal, RN, Brazil, p. 217-232, May-2003;
- Neisse, R.; Granville, L. Z.; Almeida, M. J. B.; Tarouco, L. M. R. A Web-based System to Analyze Network Management Information in XML. Proceedings of 21st Brazilian Symposium on Computer Networks (SBRC), Natal, RN, Brazil, p. 905-912 (in Portuguese), May-2003;

- Neisse, R.; Wietholter, S. Web-based Soil Analysis Quality Control Program. XIII Salao de Iniciacao Cientifica e X Feira de Iniciacao Cientifica, Porto Alegre, Rio Grande do Sul, Brazil (in Portuguese), Jan-2002;
- Neisse, R.; Pavan, W. Using Computational Reflection to Control Distributed Objects. IV Simposio de Informatica do Centro Universitario Franciscano, 1999, Santa Maria, Brazil (in Portuguese), Apr-1999;
- Neisse, R.; Pavan, W. Using Sockets and RMI to Build Distributed Systems in Java. I Simposio de Informatica do Planalto Medio (SIPM), Passo Fundo, Rio Grande do Sul, Brazil (in Portuguese), Jan-1999;
- Neisse, R.; Wietholter, S. Web-based Soil Analysis Quality Control Program. XI Mostra de Iniciacao Cientifica e III Mostra Interna de Pos-graduacao, 1999, Passo Fundo, Rio Grande do Sul, Brazil (in Portuguese), Oct-1999.

**TRUST AND PRIVACY MANAGEMENT SUPPORT FOR CONTEXT-AWARE SERVICE PLATFORMS**

*Ricardo Neisse*

In a context-aware service platform, service providers adapt their services to the current situation of the service users using context information retrieved from context information providers. In such a service provisioning platform, important trust and privacy issues arise, because different entities responsible for different tasks have to collaborate in the provisioning of the services. Context information is privacy sensitive by nature, making the communication and processing of this information a potential privacy threat.

The main goal of this thesis is to learn how to support users and providers of context-aware services in managing the trade-off between privacy protection and context-based service adaptation. More and more precise context information retrieved from trustworthy context information providers allows context-aware service provider to adapt their services more reliably. However, more and more precise context information also means a higher risk for the service users in case of a privacy violation.

UNIVERSITY OF TWENTE. CTIT SIKS

CTIT Ph.D. Thesis Series No. 11-216
ISSN 1381-3617

SIKS Dissertation Series No. 2012-09

ISBN 978-90-365-3336-2
DOI 10.3990./1.9789036533362
http://dx.doi.org/10.3990/1.9789036533362