Accounting for Values in Design
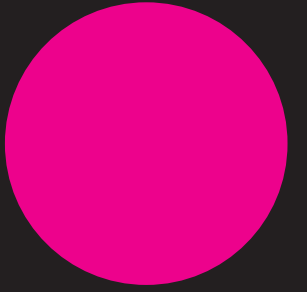
Detweiler, Christian

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Christian A. Detweiler

Accounting for Values in Design

Accounting for Values in Design

Christian A. Detweiler

Accounting for Values in Design

Accounting for Values in Design

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. ir. K.C.A.M. Luyben,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op vrijdag 28 oktober 2016 om 12:30 uur

door

Christian Alexander DETWEILER

Master of Science in Media Technology,
Universiteit Leiden, Nederland,
Bachelor of Arts in Media Studies,
Universiteit van Amsterdam, Nederland,
geboren te Morgantown, West Virginia, Verenigde Staten.

Dit proefschrift is goedgekeurd door de:

promotoren: Prof. dr. C. M. Jonker en Prof. dr. M. J. van den Hoven
copromotor: Dr. K. V. Hindriks

Samenstelling promotiecommissie:

Rector Magnificus,	voorzitter
Prof. dr. C. M. Jonker,	Technische Universiteit Delft
Prof. dr. M. J. van den Hoven,	Technische Universiteit Delft
Dr. K. V. Hindriks,	Technische Universiteit Delft

Onafhankelijke leden:

Prof. dr. D. K. J. Heylen,	Universiteit Twente
Prof. dr. ir. I. R. van de Poel,	Technische Universiteit Delft
Prof. dr. Y. Tan,	Technische Universiteit Delft
Prof. dr. M. A. Neerincx,	Technische Universiteit Delft
Prof. dr. A. Hanjalic,	Technische Universiteit Delft, reservelid

An electronic version of this dissertation is available at
http://repository.tudelft.nl/.

Contents

Summary

One of the more notable technologies to enter and affect everyday life is information and communication technology (ICT). Since the twentieth century, ICTs have had a considerable impact on many aspects of everyday life. This impact on individuals and society is rarely neutral; ICTs can have both desirable and undesirable consequences — ethical implications. One field of computing in particular envisions computing technology permeating everyday life. This field, known as Ubiquitous Computing or Pervasive Computing, aims to integrate computing technology seamlessly into the physical world and everyday life. This pervasiveness has the potential to amplify pervasive computing's ethical implications. Human values such as social well-being, privacy, trust, accountability and responsibility lie at the heart of these ethical implications. With a technology already so deeply intertwined with so many aspects of everyday life, it is increasingly important to consider the human values at stake.

This raises the question of when and by whom these values should be considered. Design plays a key part in shaping Pervasive Computing technologies' ethical implications. During design, alternatives can still be chosen, and consequences can be pondered (and pursued or avoided) before they play out. That is why it is vital to consider the values at stake in pervasive computing during design, rather than after it. In Pervasive Computing design, most attention goes toward tackling technical challenges rather than addressing ethical issues. While designers in this field may know about some of the ethical implications of pervasive computing, they often lack the training or familiarity with techniques to deal with such issues.

Over the past 20 years, Value Sensitive Design (VSD) has been developed in an attempt to provide an approach to dealing with values in design. VSD succeeds in integrating ethics with actual design in the sense that it provides techniques that help identify, conceptualize and consider values, and related concepts and issues, during the design process. With regard to design and development, VSD places considerably more emphasis on activities related to analysis than on activities related to implementation. With little concrete guidance on how to translate knowledge of values issues to technological features, there is a risk that the knowledge gained does not influence the design to the fullest possible extent.

Addressing these issues calls for an expansion of VSD's methods and techniques. More specifically, it calls for techniques to uncover and analyze the mechanisms through which Pervasive Computing's constitutive technologies (and the design alternatives that incorporate them) affect human values. The overall objective of this work is to expand VSD's conceptual, empirical and technical investigations with techniques that make explicit how Pervasive Computing technologies affect human values.

To meet this overall objective, we aim to provide techniques with which to analyze how existing Pervasive Computing technologies affect various types of values: explicitly supported values, stakeholder values and designer values. We also aim to provide techniques with which to elicit the various types of values on which these analyses focus. Finally, we aim to provide methods to capture the knowledge that results from these analyses in ways that help guide Pervasive Computing design in accounting for values.

In Chapter 2, we examine the mechanisms through which Pervasive Computing's constitutive technologies, such a sensors, activity recognition and reasoning, affect the values that Pervasive Computing designers explicitly recognize and aim to support, such as independence and safety. The guiding research questions in this work are "How do Pervasive Computing's constituent technologies affect values that are explicitly recognized in Pervasive Computing research? How can we uncover these effects?". We address these questions by presenting a literature survey of Pervasive Computing technologies for health care. We analyze the technical components and contexts of use of the applications presented in the literature we survey, as well as the values recognized in the literature. We organize this analysis in a taxonomy, and use this taxonomy to analyze the relationships between values, contexts of use and technologies in Pervasive Computing for health care. This analysis provides insight into how values that are explicitly recognized in Pervasive Computing research, such as independence, safety and privacy, are supported by some of Pervasive Computing's constituent technologies, such as activity recognition. One notable insight from this analysis is that the value of social well-being is rarely addressed in the body of literature included in the survey. We argue that understanding the analyzed relationships can help designers better consider the values their technologies affect. We believe this will help designers weave computing technologies into our everyday lives in a way that is sensitive to the needs and values of the people living those lives, and to the contexts in which they are lived.

Next, we use this knowledge of existing Pervasive Computing technologies and their effects on values to guide future design with regard to values. This is the aim of Chapter 3, where we address the question "How can recurrent value issues surrounding pervasive healthcare computing technologies be identified and used to guide design?" We present a method to discover design patterns for explicitly supported values in pervasive health computing and give several examples of patterns in Pervasive Computing for health care. We argue that design patterns that focus on values are a promising way of sharing and learning about best practices in supporting values and dealing with value issues in designing Pervasive Computing technologies for healthcare. We propose a method to derive such value sensitive design patterns from a large body of pervasive health literature and present several examples of design patterns to be used. This work does not only provide guidance in the design of a concrete system, but also enables designers and software engineers to understand how values and technology relate. We see great potential in enlarging this set of value sensitive design patterns for pervasive health computing, in putting more focus on value tensions, and ultimately developing a pattern language in the future.

While the work in Chapters 2 and 3 deal with values that are explicitly recognized and supported, Pervasive Computing technologies can affect values in ways not identified in literature. If we want to address ethical implications thoroughly during design, we cannot rely solely on value issues identified in literature. We need a way to identify and analyze further effects that Pervasive Computing technologies might have on values. The work in Chapter 4 deals with this issue, addressing the following research questions: "How do changes brought about by Pervasive Computing technologies for health care affect values? How can such effects be represented explicitly?" It examines how Ambient Intelligence systems change elderly care. We focus on the value of responsibility, analyzing how the value is instantiated in the context of elderly care, and how changes to this context through the introduction of Ambient Intelligence systems affect responsibility (and how it is instantiated). Central to our arguments is the observation that knowledge can lead to responsibilities. Because Ambient Intelligence Systems provide their users with information and support the users in using that information to extend their knowledge, the use of Ambient Intelligence Systems leads to responsibilities for both intended and unintended users. We examine different forms of responsibility and the relationship between knowledge and responsibility. We then identify knowledge and responsibilities of various stakeholders in the context of elderly care, and analyze how the introduction of Ambient Intelligence Systems affects these responsibilities by affecting what knowledge is available. We develop a semi-formal representation of stakeholders, values, context and technology, and relationships among them to support this analysis. Based on the analysis, we offer guidelines for considering identified responsibility issues during design.

Next, we turn to stakeholder values. When the values Pervasive Computing technologies implicitly or explicitly support are at odds with values held by direct and indirect stakeholders, ethical issues arise. To consider stakeholder values during design, we first need to find out what these values are. This calls for a means to elicit values, and to capture them in a form that can guide design. Chapter 5 addresses this by means of the following research questions: "How can values be elicited in the context of requirements engineering? How can values be translated into design requirements?" We present a method to extract values and related elements obtained through existing value elicitation techniques, and to document these elements in a format that is amenable to use in further requirements engineering activities, as an approach to integrate values-oriented techniques with a requirements engineering process.. We present a case study as a proof of concept. In the case study, we demonstrate that our approach yields value stories – user stories, a concept familiar in requirements engineering, that include values. Our approach helps requirements engineers and stakeholders identify values and translate them into a form that is suitable for use in the further requirements engineering process.

In the previous chapters, we develop ways of analyzing relationships between (explicitly supported) values, contexts and technologies and making these relationships explicit. Understanding these relationships, we argue, is a necessary step to address value issues during design. However, this step only tackles the issue of

identifying and analyzing value issues, and leaves open the issue of incorporating these analyses in a design process. VSD offers limited support in this regard, though it does so purposefully, as it is intended to be combined with other design methods. This raises the issue of which design methods, exactly, it should be combined with and whether knowledge on values can readily be used in those methods. That is, it is unclear whether design methods outside of the VSD framework can readily incorporate concepts from VSD. Some requirements and software engineering concepts seem similar to values. However, there are some important differences between values and these concepts. Many software design methods address "soft issues", but very few address values explicitly.

Chapter 6 deals with making values explicit in existing software design methods. It addresses the following questions: "Are there concepts in software design that could potentially capture value knowledge obtained through conceptual investigations? What are the similarities and differences between these concepts and values? What should a design process adhere to if it is to incorporate knowledge about values?" We present six principles that design methods should meet in order to properly deal with values. One area in which adherence to stakeholder values is important, is Agent-Oriented Software Engineering (AOSE). The Tropos AOSE method, with its concept of soft-goal, comes close to meeting our principles, but does not address values explicitly. We discuss a case study in which we attempt to capture values in Tropos' soft-goals after eliciting them using techniques from Value-Sensitive Design. We show that Tropos cannot fully handle the six principles we propose. We argue that Tropos' soft-goals are fundamentally different from human values as studied here. To address these problems, we propose complementing Tropos with a separate, first-class entity to capture values. This entity will allow the designer to explicitly represent values throughout the design process, and to make values concrete enough to operationalize them and to expose and address conflicts between them.

Beside eliciting stakeholder values in the requirements engineering phase of new technologies, we need to be able to examine how existing technologies support stakeholders' values. Chapter 7 aims to do this and addresses the following research questions: "How do designer values and stakeholder values in Pervasive Computing applications for personal well-being affect acceptance? Do similarities in designer and stakeholder values impact how stakeholders experience the technology? Does this influence stakeholders' acceptance of technology?" In Chapter 7, we investigate the role of user personality and values in relation to factors influencing the use of Pervasive Computing applications for healthy food consumption. This research is motivated by the observation that these applications employ diverse strategies, which are more or less prescriptive, to support healthy eating behaviors. The choice of strategy reflects the designer's stance on how behavior change can be achieved. Judgments on values such as autonomy underlie these stances and we argue that an alignment of user values and personality and designer stances will lead to more successful app use and, ultimately, sustainable health behaviors. We conduct a pilot study, in the form of an online experiment, to investigate the relationship of personality, values and several factors related to the use of apps that employ

strategies ranging from little to very prescriptive. With online questionnaires we measure the personality traits and values of each participant as well as several factors related to each app, such as intention to use, usefulness, effort, feeling, etc. We select three apps for tracking food consumption (Evernote Food, Foodzy and Noom Coach). Our aim is to identify one app that is strongly prescriptive, and one that is at the opposite end of the prescriptiveness spectrum (i.e., effectively open-ended), and one that falls in the middle.

The results show that apps were rated significantly different. However, so far few correlations between value and personality factors and the intention to use an app have been found. Most prominent is that the intention to use Evernote, the least prescriptive app, was rated low all in all groups of people. This may be based on the low perceived usefulness. The fact that hedonism was negatively correlated with Evernote hints at that participants did not perceive using this app as fun. Interestingly, people valuing power seemed to trust the apps less. Although these preliminary results cannot be used for drawing general conclusions, they suggest that research should be devoted to design for trustworthiness and clear goals (missing in Evernote). Transparency is key in both to allow users to understand the underlying stances of an app and the fit to the users' attitude.

Reflecting on this work, its components appear to fall into three broad categories of activity: identifying and examining sources of values, analyzing uncovered values, and capturing analyses in artifacts. These three general activities form a process. In this thesis, we follow five different paths through this process from source, via analysis, to artifact.

Other paths are yet to be explored. For example, our Value Sensitive Design Patterns use explicitly supported values as a source, but another important source to examine is stakeholder values. Similarly, the Value Stories capture stakeholder values. It would be interesting to use Value Stories to elicit and capture designers' values. A related direction for future work is to study this overall process and the interactions between (the outcomes of) the various activities. Some artifacts that result from the process can serve as the source for further iterations. For example, Value Stories could form the basis for Value-Sensitive Design Patterns. Future work should also study and evaluate these methods in use in actual design processes, from early phases of design through implementation. It should examine the advantages and difficulties of applying these methods in a design process. Ultimately, it should assess to what extent these methods help incorporate human values in design.

Samenvatting

Een van de meer noemenswaardige technologieën die zijn intrede in het dagelijks leven heeft gedaan en deze beïnvloedt is Informatie- en Communicatietechnologie (ICT). Sinds de twintigste eeuw hebben ICTs een aanzienlijke invloed gehad op vele aspecten van het dagelijks leven. Deze invloed op individuen en de samenleving is zelden neutraal; ICTs kunnen zowel wenselijke als onwenselijke gevolgen hebben - ethische implicaties. Eén gebied van informatica in het bijzonder stelt zich een wereld voor waarin ICT in het dagelijks leven doordringt. Dit gebied, Ubiquitous Computing of Pervasive Computing genaamd, streeft ernaar ICTs naadloos in de fysieke omgeving en het dagelijks leven te integreren. Deze alomtegenwoordigheid (pervasiveness in pervasive computing) heeft de potentie om de ethische implicaties van Pervasive Computing te versterken. Menselijke waarden als sociaal welzijn, privacy, vertrouwen, toerekenbaarheid en verantwoordelijkheid liggen aan deze ethische implicaties ten grondslag. Daarom, met een technolgie die nu al zo nauw verweven is in zo veel aspecten van ons dagelijks leven, wordt het steeds belangrijker om rekening te houden met de menselijke waarden van belang.

Dit roept de vraag op wanneer en door wie deze waarden in aanmerking genomen moeten worden. Ontwerp speeld een sleutelrol bij de vorming van ethische implicaties van Pervasive Computing. Tijdens het ontwerpproces kan er nog gekozen worden tussen verschillende alternatieven en de consequenties kunnen worden overdacht (en nagestreefd of vermeden) voordat ze realiteit worden. Daarom is het essentieel om de waarden van belang te beschouwen tijdens het ontwerpproces, in plaats van na die tijd. In Pervasive Computing ontwerp gaat de meeste aandacht naar het oplossen van technische problemen in plaats van ethische kwesties. Hoewel ontwerpers in dit vakgebied kennis hebben over sommige van de ethische implicaties van Pervasive Computing, hebben ze vaak niet voldoende training in technieken om met deze problemen om te gaan.

In de afgelopen 20 jaar is Value Sensitive Design (VSD) ontwikkeld in een poging om een manier te bieden om in het ontwerpproces met waarden om te gaan. VSD slaagt er in om ethiek te integreren met ontwerp in de zin dat het technieken biedt die helpen tijdens het ontwerpproces waarden en verwante concepten en kwesties te identificeren, conceptualiseren en overwegen. Met betrekking tot ontwerpen en ontwikkelen legt VSD meer nadruk op analyseactiviteiten dan implementatieactiviteiten. Zonder concrete richtlijnen om kennis over waardenkwesties in technologische functies te vertalen bestaat er een risico dat de verworven kennis het ontwerp niet zo volledig mogelijk beïnvloedt.

Het aanpakken van deze problemen vraagt om een uitbreiding van de methoden en technieken die VSD biedt. In het bijzonder vraagt het om technieken om de mechanismen waarmee de constitutieve technologieën van Pervasive Computing waarden aantasten te ontdekken en te analyseren. De overkoepelende doelstel-

ling van dit werk is om VSDs conceptuele, empirische en technische onderzoeken (conceptual, empirical en technical investigations) uit te breiden met technieken die expliciet maken hoe Pervasive-Computingtechnologieën menselijke waarden beïnvloeden.

Om aan deze overkoepelende doelstelling te voldoen streven we ernaar technieken te verschaffen waarmee geanalyseerd kan worden hoe bestaande Pervasive-Computingtechnologieën effect hebben op verschillende soorten waarden: expliciet ondersteunde waarden, waarden van belanghebbenden (stakeholders) en waarden van ontwerpers. We streven er ook naar technieken aan te bieden waarmee deze verschillende soorten waarden kunnen worden ontlokt. Tenslotte streven we ernaar methoden aan te bieden waarmee de kennis uit deze analyses kan worden vastgelegd op manieren die ondersteuning biedt in het rekening houden met waarden tijdens het ontwerp van Pervasive Computingsystemen.

In Hoofdstuk 2 bestuderen we de mechanismen waarmee de constitutieve technologieën van Pervasive Computing, zoals sensoren, activiteitsherkenning en redeneren, de waarden die ontwerp van Pervasive-Computingtechnologieën expliciet erkennen en pogen te ondersteunen, zoals onafhankelijkheid en veiligheid. De leidende onderzoeksvragen in dit werk zijn "Hoe beïnvloeden de samenstellende technologieën van Pervasive Computing de waarden die expliciet worden erkend in onderzoek naar Pervasive Computing? Hoe kunnen deze effecten worden ontdekt?" Wij behandelen deze vragen door een literatuuronderzoek naar Pervasive-Computingtechnologieën voor gezondheidszorg te presenteren. We analyseren de technische componenten en gebruikscontexten van de applicaties die in de onderzochte literatuur worden gepresenteerd, evenals de waarden die in de literatuur wordengenoemd. We structuren deze analyse in een taxonomie, en gebruiken deze taxonomie om de relaties tussen waarden, gebruikscontexten en technologieën in Pervasive Computing voor gezondheidszorg te analyseren. Deze analyse geeft inzicht in hoe expliciet erkende waarden in onderzoek naar Pervasive Computing, zoals onafhankelijkheid, veiligheid en privacy, ondersteund worden door sommige van de samenstellende technologieën van Pervasive Computing voor gezondheidszorg, zoals activity recognition. Een noemenswaardig inzicht uit deze analyse is dat de waarde sociaal welzijn zelden wordt geadresseerd in de onderzochte literatuur. Wij stellen dat het begrijpen van de geanalyseerde relaties ontwerpers kan helpen om beter rekening te houden met de waarden die hun technologieën kunnen beïnvloeden. Wij menen dat dit ontwerpers zal helpen ICTs in ons dagelijks leven te verweven op een manier die rekening houdt met de behoeften en waarden van degenen in wiens leven ze worden verweven, en de contexten waarin dit gebeurt.

Vervolgens gebruiken we deze kennis van bestaande Pervasive-Computingtechnologieën en de effecten daarvan op waarden om toekomstig ontwerp richtlijnen te bieden met betrekking tot waarden. Dit is het doel van Hoofdstuk 3, waar we de volgende vraag adresseren: "Hoe kunnen terugkerende waardenkwesties rondom Pervasive-Computingtechnologieën voor gezondheidszorg worden geïdentificeerd en gebruikt als richtlijn voor ontwerp?" We presenteren een methode om design patterns voor expliciet ondersteunde waarden in deze technologieën te ontdekken,

en geven een aantal voorbeelden van zulke design patterns. Wij betogen dat design patterns die waarden centraal stellen een veelbelovende manier zijn om best practices in het ondersteunen van waarden en omgaan met waardenkwesties in het ontwerp van Pervasive-Computingtechnologieën voor gezondheidszorg te delen en te leren. Wij stellen een methode voor om zulke Value Sensitive Design Patterns te ontlenen aan een grote verzameling literatuur op het gebied van Pervasive-Computing voor gezondheidszorg en presenteren meerdere voorbeelden van het gebruik van zulke design patterns. Dit werk biedt niet alleen richtlijnen bij het ontwerp van een concreet systeem, maar stelt ontwerpers en software engineers ook in staat om te begrijpen hoe waarden en technologie zich tot elkaar verhouden. Wij zien er veel potentie in deze verzameling Value Sensitive Design Patterns voor Pervasive Computing voor gezondheidszorg uit te breiden, meer nadruk te leggen op value tensions en uiteindelijk een pattern language te ontwikkelen.

Hoewel het werk in Hoofdstukken 2 en 3 gaat over waarden die expliciet worden erkend en ondersteund kunnen Pervasive-Computingtechnologieën ook waarden beïnvloeden op manieren die niet in de literatuur worden geïdentificeerd. Als het doel is om ethische implicaties grondig te adresseren gedurende het ontwerpproces volstaat het niet om enkel te steunen op waardenkwesties die in de literatuur worden aangekaart. Er is een manier nodig om verdere effecten die Pervasive-Computingtechnologieën op waarden kunnen hebben te identificeren en analyseren. Het werk in Hoofdstuk 4 adresseert dit probleem, en behandelt daarbij de volgende onderzoeksvragen: "Hoe kunnen veranderingen die teweeg worden gebracht door Pervasive-Computingtechnologieën voor gezondheidszorg waarden aantasten? Hoe kunnen zulke effecten expliciet worden gerepresenteerd?" Dit werk onderzoekt hoe Ambient Intelligence systemen ouderenzorg veranderen. Wij richten ons daarbij op de waarde verantwoordelijkheid en analyseren hoe de waarde wordt geconcretiseerd in de context van ouderenzorg, en hoe veranderingen aan deze context door het introduceren van Ambient-Intelligencesystemen verantwoordelijkheid (en hoe het wordt geconcretiseerd) beïnvloedt. In onze argumenten staat de observatie dat kennis tot verantwoordelijkheden kan leiden centraal. Omdat Ambient-Intelligencesystemen hun gebruikers van informatie voorzien en deze gebruikers helpen hun kennis uit te breiden, leidt het gebruik van Ambient-Intelligencesystemen tot verantwoordelijkheden voor zowel bedoelde als onbedoelde gebruikers. Wij onderzoeken verschillende vormen van verantwoordelijkheid en de relatie tussen kennis en verantwoordelijkheid. Vervolgens identificeren wij de kennis en verantwoordelijkheden van verschillende belanghebbenden in de context van ouderenzorg, en analyseren wij hoe het introduceren van Ambient-Intelligencesystemen deze verantwoordelijkheden verandert door te beïnvloeden welke kennis beschikbaar is. Om deze analyse te ondersteunen ontwikkelen wij een semi-formele representatie van stakeholders, waarden, context en technologie, en de relaties daartussen. Op basis van de analyse bieden wij richtlijnen om tijdens het ontwerp rekening te houden met de geïdentificeerde kwesties rondom verantwoordelijkheid.

Vervolgens wenden we ons tot de waarden van belanghebbenden (stakeholders). Wanneer de waarden die Pervasive-Computingtechnologieën impliciet of

expliciet ondersteunen ingaan tegen de waarden van directe en indirecte belang-
hebbenden, ontstaan er ethische problemen. Om tijdens het ontwerpproces re-
kening te houden met de waarden van belanghebbenden moet eerst bekend zijn
welke waarden dat zijn. Dit vraagt om een middel om waarden te eliciteren en
deze vast te leggen in een vorm die houvast kan bieden bij ontwerp. Hoofdstuk
5 adresseert dit met de volgende onderzoeksvragen: "Hoe kunnen waarden wor-
den geëliciteerd in de context van requirements engineering? Hoe kunnen waarden
worden vertaald naar requirements voor ontwerp?" Wij presenteren een methode
om waarden en verwante elementen (die verkregen zijn uit bestaande technieken
om waarden te eliciteren) te documenteren in een vorm die zich leent voor gebruik
in verdere Requirements-Engineeringactiviteiten. Dit is een manier om waarden-
georiënteerde technieken te integreren in een Requirements-Engineeringproces.
Wij presenteren een casus als proof of concept. In de casus tonen we aan dat onze
aanpak Value Stories oplevert - user stories, een bekend concept in Requirements
Engineering, die waarden bevatten. Onze aanpak helpt Requirements Engineers en
belanghebbende om waarden te identificeren en deze te vertalen naar een vorm
die geschikt is voor gebruik in het verdere Requirements-Engineeringproces.

In de voorgaande hoofdstukken ontwikkelden we manieren om relaties tussen
(expliciet ondersteunde) waarden, contexten en technologieën te analyseren en
om deze relaties expliciet te maken. Wij betogen dat begrip van deze relaties een
noodzakelijke stap is in het adresseren van waardekwesties gedurende het ont-
werpproces. Echter, deze stap omvat alleen het identificeren en analyseren van
waardekwesties en laat het probleem van het opnemen van deze analyses in een
ontwerpproces open. VSD biedt beperkte ondersteuning in dit opzicht, weliswaar
met opzet, omdat het bedoeld is om met andere ontwerpmethoden gecombineerd
te worden. Dit roept wel de vraag op met welke ontwerpmethoden het dan gecom-
bineerd zou moeten worden en of kennis over waarden gemakkelijk in die methoden
kan worden gebruikt. Dat wil zeggen, het is onduidelijk of ontwerpmethoden bui-
ten het VSD-raamwerk gemakkelijk concepten uit VSD in zich op kunnen nemen.
Concepten uit sommige Requirements- en Software-Engineeringmethoden lijken op
het eerste gezicht op waarden. Echter, er zijn belangrijke verschillen tussen waar-
den en deze concepten. Vele software-ontwerpmethoden adresseren "soft issues",
maar weinig adresseren expliciet waarden.

Hoofdstuk 6 gaat over het expliciet maken van waarden in bestaande software-
ontwerpmethoden. Dit werk adresseert de volgende onderzoeksvragen: "Zijn er
concepten in softwareontwerp die kennis over waarden die verkregen is uit con-
ceptual investigations zouden kunnen vastleggen? Wat zijn de overeenkomsten en
verschillen tussen deze concepten en waarden? Waaraan zou een ontwerpproces
zich moeten houden om kennis over waarden te incorporeren?" Wij presenteren
zes principes waaraan ontwerpmethoden zouden moeten voldoen om adequaat met
waarden om te gaan. Een gebied van softwareontwerp waarin naleving van waar-
den van belanghebbenden belangrijk is, is Agent-Oriented Software Engineering
(AOSE). De Tropos AOSE-methode voldoet met het concept van soft-goal bijna aan
de principes, maar adresseert waarden niet expliciet. Wij bespreken een case study
waarin we pogen in Tropos waarden vast te leggen met soft-goals nadat wij deze

eliciteren met technieken uit VSD. Wij tonen aan dat Tropos niet volledig in staat is aan de zes voorgestelde principes te voldoen. Wij stellen dat soft-goals in Tropos fundamenteel verschillen van menselijke waarden zoals deze hier worden onderzocht. Om deze problemen aan te pakken, stellen wij voor Tropos aan te vullen met een aparte first-class entiteit om waarden te representeren. Deze entiteit stelt de ontwerper instaat waarden expliciet door het hele ontwerproces heen te representeren en waarden voldoende concreet te maken om deze te operationaliseren, bloot te leggen, en conflicten tussen waarden te adresseren.

Naast het eliciteren van waarden van belanghebbenden in de Requirements-Engineeringfase van het ontwerp van nieuwe technologieën moet er ook onderzocht kunnen worden hoe bestaande technologieën waarden van belanghebbenden ondersteunen. Hoofdstuk 7 streeft dit na en adresseert de volgende onderzoeksvragen: "Hoe beïnvloeden de waarden van ontwerpers en belanghebbenden in Pervasive-Computingapplicaties voor persoonlijk welzijn de acceptatie van die applicaties? Beïnvloeden overeenkomsten tussen waarden van ontwerpers en belanghebbenden hoe belanghebbenden de technologie ervaren? Heeft dit invloed op de mate waarin belanghebbenden de technologie accepteren? In Hoofdstuk 7 onderzoeken we de rol van de persoonlijkheid en waarden van gebruikers in relatie tot andere factoren die invloed hebben op het gebruik van Pervasive-Computingapplicaties voor gezonde voeding. De aanleiding voor dit onderzoek is de observatie dat deze applicaties verschillende strategieën inzetten om gezond eetgedrag te ondersteunen die meer of minder voorschrijvend zijn. De strategiekeuze weerspiegelt het standpunt dat de ontwerper inneemt over hoe gedragsverandering kan worden bewerkstelligd. Oordelen over waarden als autonomie liggen aan deze standpunten ten grondslag en wij stellen dat het op één lijn brengen van waarden en persoonlijkheid van gebruikers met standpunten van ontwerpers zal leiden tot succesvollere applicatiegebruik en uiteindelijk tot duurzaam gedrag met betrekking tot gezondheid. Wij voeren een pilotstudie uit in de vorm van een online experiment om de relatie te onderzoeken tussen persoonlijkheid, waarden en verschillende factoren die te maken hebben met het gebruik van applicaties die strategieën inzetten die variëren van weinig tot zeer voorschrijvend. Met online enquêtes meten we de karaktereigenschappen en waarden van elke deelnemer, evenals verschillende applicatiegerelateerde factoren als intentie om te gebruiken, nut, inspanning, gevoel, etc. We selecteren drie applicaties voor het bijhouden van voedselconsumptie (Evernote Food, Foodzy en Noom Coach). Ons doel is een applicatie te identificeren die sterk voorschrijvend is, één die geheel niet voorschrijvend is, en één die ertussenin valt.

De resultaten van ons onderzoek tonen aan dat de applicaties significant anders zijn beoordeeld. Echter, vooralsnog zijn weinig correlaties gevonden tussen waarden, persoonlijkheidsfactoren en intentie tot gebruik. Het meest opvallende is dat de intentie om Evernote te gebruiken, de minst voorschrijvende applicatie, laag scoorde in alle groepen deelnemers. Dit heeft mogelijk te maken met het lage ervaren nut van de applicatie. Het feit dat hedonisme negatief correleerde met Evernote suggereert dat deelnemers de applicatie als weinig plezierig ervoeren. Mensen voor wie macht een belangrijke waarde was leken de applicaties minder

te vertrouwen. Hoewel op basis van deze preliminaire resultaten geen algemene conclusies kunnen worden getrokken, suggereren ze dat onderzoek zich moet richten op betrouwbaarheid en duidelijke doelstellingen (allebei in Evernote afwezig). Transparantie is belangrijk, zowel om gebruikers in staat te stellen om de onderliggende standpunten van een applicatie te begrijpen als om overeen te stemmen met de houding van de gebruiker.

Reflecterend op het werk in dit proefschrift zijnde onderdelen ervan in drie brede categorieën van activiteit onder te verdelen: het identificeren en onderzoeken van bronnen van waarden, het analyseren van ontdekte waarden, en het vastleggen van waarden in artefacten. Deze drie activiteiten vormen een proces. In dit proefschrift volgen we vijf verschillende paden voor het proces van bron, via analyse naar artefact.

Andere paden moeten nog onderzocht worden. Zo nemen bijvoorbeeld onze Value Sensitive Design Patterns expliciet ondersteunde waarden als bron, maar een andere belangrijke bron om in deze context te onderzoeken is de waarden van belanghebbenden. Vergelijkbaar leggen Value Stories waarden van belanghebbenden vast, en zou het interessant zijn om Value Stories ook te gebruiken om waarden van ontwerpers te eliciteren en vast te leggen. Een verwante richting voor toekomstig werk is het bestuderen van het gehele proces en de interacties tussen de (uitkomsten van de) verschillende activiteiten. Sommige artefacten die uit het proces voortvloeien kunnen als basis dienen van verdere iteraties. Zo zouden bijvoorbeeld Value Stories de basis kunnen vormen van Value Sensitive Design Patterns. Toekomstig werk zou deze methoden ook in de praktijk moeten onderzoeken en evalueren tijdens het gebruik in ontwerpprocessen, van vroege ontwerpfase tot implementatie. Het zou de voordelen en uitdagingen moeten onderzoeken van het toepassen van toepassen van deze methoden in een ontwerpproces. Uiteindelijk zou dit onderzoekmoeten bepalen in welke mate deze methoden helpen menselijke waarden mee te nemen in ontwerp.

1

Introduction to this Dissertation

This chapter presents the topic and key concepts of this thesis, discusses the main problems on which the thesis centers, and outlines the research objectives and questions the thesis aims to answer.

1

1.1. Introduction

Technological artifacts have ethical implications. Technological artifacts or tools enter people's everyday lives, extending existing human capabilities and introducing new ones. By providing these capabilities, technological artifacts have the potential to change how people achieve their goals and even the goals they are able to achieve. Technological artifacts can and often do affect the contexts of their use. Examples abound. Rail transport reduced the time it took to travel great distances, increasing the mobility of people and goods. In the United States, rail transport changed time-keeping itself [1]. The telephone let people communicate directly over long distances. The phonograph introduced the ability to record and reproduce sounds, which changed how music was heard. Technological artifacts' imjpact on individuals and society is rarely neutral; they can have both desirable and undesirable consequences — ethical implications.

One of the more notable technologies to enter and affect everyday life is information and communication technology (ICT). Since the twentieth century, ICTs have had a considerable impact on many aspects of everyday life. One field of computing in particular envisions computing technology permeating everyday life. This field, known as ubiquitous computing or pervasive computing, aims to integrate computing technology seamlessly into the physical world and everyday life, making it disappear into the background. Thanks to advances in mobile computing, embedded systems, and distributed computing, and the declining price and size of processors, we are increasingly living in the world ubiquitous computing envisioned. Computing *is* pervasive in our smartphones, tablets, and many other visible and less visible devices and infrastructure.

This pervasiveness has the potential to amplify pervasive computing's ethical implications. For example, mobile telephony and computing have given people with access to these technologies the new capabilities to communicate at distance. By removing some of the barriers of physical distance, mobile telephony has arguably benefited its users' connectedness. At the same time, mobile telephony and computing pose threats to people's privacy, as illustrated by recent news that the United States' National Security Agency collects as many as 5 billion records daily to track mobile phones across the globe [2]. Human values such as social well-being, privacy, trust, accountability and responsibility lie at the heart of these ethical implications. With a technology already so deeply intertwined with so many aspects of everyday life, it is increasingly important to consider the human values at stake.

This raises the question of when and by whom these values should be considered. Like other technological artifacts, pervasive computing technologies and their effects do not just appear in society. They are the result of a process that involves deliberate activities and decisions — the design process. In this process, designers and other stakeholders determine which needs or problems their design should address. Designers set the scope of their solution, and make decisions between alternatives to realize this solution. Design plays a key part in shaping technology's ethical implications. Designed differently, a pervasive computing technology would have different ethical implications. During design, alternatives can still be chosen, and consequences can be pondered (and pursued or avoided) before they play out.

That is why it is vital to consider the values at stake in pervasive computing during design, rather than after it.

1.2. Values in Design

The view that technology has ethical implications, and that these are influenced to an important extent by design, has not always been widely held. Until the second half of the twentieth century, an instrumental perspective of technology was more common. On this view, technology has the generally desirable effect of increasing human capabilities. Where technology does have undesirable consequences, the instrumental perspective attributes these to the people using the technology, and considers the technology neutral [3]. From the second half of the twentieth century, more emphasis was placed on the choices made in technological development. Specific technologies and specific stages of technological development became the focus of ethical reflection on technology, rather than technology as such [3]. Computing technology is one of the specific technologies that became the focus of ethical reflection in computer ethics.

As a field, computer ethics emerged in the second half of the 1970s with the work of Walter Maner, Deborah Johnson and James Moor, among others [4], though Norbert Wieners' work on cybernetics in the 1940s and later envisioned ethical implications of computing technology [4]. The work in computer ethics beginning in the 1970s was characterized by ethical reflection and often aimed at providing educational materials [4]. This emphasis on ethical reflection limits computer ethics' usefulness to design, as it leaves computer ethics too far removed from technical implementations [5].

Other work related to computer ethics focused on actively protecting human values. In *The Human Use of Human Beings*, Norbert Wiener examined potential harms and benefits of introducing cybernetic machines into society, and often discussed ways to protect or advance human values, such as life, health, security, knowledge, opportunity, ability, democracy, happiness, peace and freedom [6]. Subsequent approaches that deal with human values include Social Informatics, Computer-Supported Collaborative Work and Participatory Design, though each faces limitations in integrating ethics with design [5].

More recently, Value Sensitive Design (VSD) has emerged in an attempt to overcome some of the challenges in integrating ethics with design. The VSD framework offers an iterative and integrative approach to accounting for values in design, through conceptual, empirical and technical investigations. Conceptual investigations identify direct and indirect stakeholders affected by the design, and values that are implicated. Empirical investigations examine actual or potential stakeholders and contexts-of-use, potentially using quantitative and qualitative methods from social science research. Technical investigations aim to design new technology to support selected values or examine how existing technologies support or hinder certain values [7].

Throughout these investigations, VSD draws on a number of key concepts. It distinguishes between direct stakeholders, who directly interact with technology, and indirect stakeholders who are affected by the technology or its output with-

out interacting with it. Values are another key concept, which work on VSD has defined as "what a person or group of people considers important in life" [7]. VSD adopts an interactional stance towards values, which considers values neither solely determined by technology nor by social systems, but by a combination of the two. VSD distinguishes between three types of values. Explicitly supported values are those that are required to be supported by the technology. Designer values refer to the designers or researchers personal values that implicitly guide design decisions. Stakeholder values are values that are important to some, though not necessarily all, stakeholders of a technology [8]. Differences among values can give rise to value tensions (such as privacy versus security). In some work in VSD (for example, [5], a list of "values with ethical import" has been presented as a heuristic list to suggest values to consider, consisting of human welfare, ownership and property, privacy, freedom from bias, universal usability, trust, autonomy, informed consent, accountability, identity, calmness, and environmental sustainability.

VSD succeeds in integrating ethics with actual design in the sense that it provides techniques that help identify, conceptualize and consider values, and related concepts and issues, during the design process. Nevertheless, a number of researchers have identified limitations of VSD. It has been argued that VSD's use of a list of twelve values of ethical import [7] as a heuristic puts a discursive definition of values before values that could be discovered through investigation, favors known values to value discovery, and leads to systems that align with those twelve values rather than with values expressed in the context of design [9]. Furthermore, it is argued that VSD does not provide guidance on which empirical instruments to use in its empirical investigations [9]. Other work argues that VSD's claim that certain values are universally held can have undesirable consequences [10]. This work also points out that VSD's heuristic lists of values do not state the cultural context in which they were developed, which can lead to implicit claims of more universality of the listed values than warranted [10]. Furthermore, there are problems with how VSD replaces participants' voices with the researcher's voice in presenting qualitative results, in that it overclaims the researcher's knowledge and authority [10]. It is also argued that the researcher is often less visible in VSD writing, which obscures the researcher's background, relation to participants in the study and the role of the researcher's own values, which again implies more authority and impartiality than warranted [10].

Most of these concerns are related more to ethics than to technology design and development, with the exception of VSD's lack of guidance on empirical investigations. With regard to design and development, we argue that work on VSD offers few suggestions regarding implementation activities. That is, work on VSD places considerably more emphasis on activities related to analysis, in both quantity and detail of suggested activities, than on activities related to implementation. For example, of the ten "practical suggestions for using Value Sensitive Design" offered in [11], only one, "Heuristics for Technical Investigations", specifically targets implementation. We regard this emphasis on analytical activities and relative lack of guidance on implementation as a challenge to properly account for values in design. Though VSD may yield knowledge about values in a prospective technology's con-

1

text of use, if there is no concrete guidance on how to translate this knowledge to technological features, there is a risk that the knowledge gained does not influence the design to the fullest possible extent.

1.3. Pervasive Computing

At the core of ubiquitous computing lies a vision in which computing technology is available to anyone, anywhere at any time (or everywhere all the time), as computers move away from the desktop, disappear into the background, and become intertwined with our everyday lives. People will no longer be forced to interact with computers behind a desk using a mouse, keyboard and a screen because computers will be available throughout the physical environment and will offer new modalities for interaction. This paradigm does raise issues and present challenges both to those using and those designing ubiquitous computing technology.

Understanding the impact that this technology has or will have is a challenge. As this technology becomes more deeply intertwined with people's everyday lives, its potential to affect those everyday lives grows. It increasingly has a hand in shaping people's actions, decisions and practices, by providing functions and features that support certain activities, and not others. In doing so, it supports certain human values more readily than others, and potentially hinders some values. Ubiquitous computing's key aim of making technology effectively invisible poses a potential challenge to observing and understanding the impact it has.

This is also a challenge to designing ubiquitous computing technology. Designing technology that is successful in becoming interwoven with people's everyday lives requires more than focus on technical issues. The designer needs to take into account who the stakeholders of her design are, what their needs are, and how the design will affect them. The designer also needs to take into account the aspects of people's everyday lives that will affect how and under which circumstances the technology is used. More importantly, considering the extent these technologies' intertwining with everyday life, the designer should consider ethical issues, such as which values the technology supports or could otherwise affect and whose values they are. Designers will have to address these recurrent issues time and again.

1.4. Research Objectives

As computing technologies' pervasiveness grows, so will the scale of their ethical implications. Rather than wait for such issues to reach their full scale, we aim to anticipate and address these issues during new pervasive computing technologies' design. Pervasive computing is an emerging technology. As such, the design decisions that shape it are heavily influenced by technical opportunities and challenges. These technology-focused decisions have a bearing on Pervasive Computing technologies' ethical implications. To address Pervasive Computing's ethical implications, we need to consider values within this decision making process.

In design processes that focus on technical innovation, as many in Pervasive Computing do, most attention goes toward tackling technical challenges rather than addressing ethical issues. While designers in this field may know about some of the

1

ethical implications of pervasive computing, they often lack the training or familiarity with techniques to deal with such issues. VSD offers such techniques. However, much of the work on VSD falls within the field of Human-Computer Interaction, and it shares much of its focus on design with that field. While Human-Computer Interaction is certainly an important aspect of designing Pervasive Computing technologies, many of the ethical implications of these technologies stem from decisions made before the user and context of use are in the picture. Most work on VSD to date does not focus on the level of technical detail that characterizes the design of Pervasive Computing technologies.

Addressing these issues calls for an expansion of VSD's methods and techniques. More specifically, it calls for techniques to uncover and analyze the mechanisms through which Pervasive Computing's constitutive technologies (and the design alternatives that incorporate them) affect human values. The overall objective of this work is to expand VSD's conceptual, empirical and technical investigations with techniques that make explicit how Pervasive Computing technologies affect human values.

To meet this overall objective, we aim to provide techniques with which to analyze how existing Pervasive Computing technologies affect various types of values: explicitly supported values, stakeholder values and designer values. We also aim to provide techniques with which to elicit the various types of values on which these analyses focus. Finally, we aim to provide methods to capture the knowledge that results from these analyses in ways that help guide Pervasive Computing design.

1.5. Dissertation Outline

Our first analysis technique examines the mechanisms through which Pervasive Computing's constitutive technologies, such a sensors, activity recognition and reasoning, affect the values that Pervasive Computing designers explicitly recognize, such as independence and safety.

Research questions Which values are explicitly supported by Pervasive Computing applications for elderly care? How do this application domain's constituent technologies achieve that support?

In Chapter 2, we address this question by presenting a literature survey of Pervasive Computing technologies for health care. We analyze the technical components and contexts of use of the applications presented in the literature we survey, as well as the values recognized in the literature. We organize this analysis in a taxonomy, and use this taxonomy to analyze the relationships between values, contexts of use and technologies in Pervasive Computing for health care.

Next, we aim to use knowledge of existing Pervasive Computing technologies and their effects on values to guide future design with regard to values.

Research questions How can recurrent value issues surrounding pervasive computing applications for health care be identified, analyzed and used to guide design?

We address this question in Chapter 3.We present a method to discover design patterns for explicitly supported values in pervasive health computing and give several examples of patterns in Pervasive Computing for health care.

Having dealt with discovery and analysis of relationships between Pervasive Computing technologies for health care, and how such analysis can guide design, we turn to values that are affected but not explicitly recognized. Pervasive computing technologies can affect values in ways not identified in literature. If we want to address ethical implications thoroughly during design, we cannot rely solely on value issues identified in literature. We need a way to identify and analyze further effects pervasive computing technologies might have on values.

Research questions How do pervasive computing technologies for elderly care affect values that are not explicitly supported? How can such effects be identified, analyzed and made explicit?

Chapter 4 focuses on these questions. It examines how Ambient Intelligence systems change elderly care. We focus on the value of responsibility, analyzing how the value is instantiated in the context of elderly care, and how changes to this context through the introduction of Ambient Intelligence systems affect responsibility (and how it is instantiated). We develop a semi-formal representation of stakeholders, values, context and technology, and relationships among them to support this analysis.

Finally, we focus on stakeholder values. When the values Pervasive Computing technologies implicitly or explicitly support are at odds with values held by direct and indirect stakeholders, ethical issues arise. For example, a technology that supports openness might be at odds with its users' privacy. To consider stakeholder values during design, we first need to find out what these values are. This calls for a way to elicit values, and capture them in a form that can guide design.

Research questions How can values be elicited within a Requirements Engineering process? How can elicited values be translated into design requirements?

In Chapter 5, we present a method to extract values and related elements obtained through existing value elicitation techniques, and to document these elements in a format that is amenable to use in further requirements engineering activities. We present a case study as a proof of concept.

In the previous chapters, we develop ways of analyzing relationships between (explicitly supported) values, contexts and technologies and making these relationships explicit. Understanding these relationships, we argued, is a necessary step to address value issues during design. However, this step only tackles the issue of identifying and analyzing value issues, and leaves open the issue of incorporating these analyses in a design process. VSD offers limited support in this regard, though it does so purposefully, as it is intended to be combined with other design methods. This raises the issue of which design methods, exactly, it should be combined with and whether knowledge on values can readily be used in those methods.That is, it is unclear whether design methods outside of the VSD framework can readily incorporate concepts from VSD. Many software design methods address "soft issues", but very few address values explicitly.

Research questions What concepts in software design could potentially capture
value knowledge? What are the similarities and differences between these
concepts and values? What should a design process adhere to if it is to
incorporate knowledge about values?

We address these questions in Chapter 6. We present six principles that design
methods should meet in order to properly deal with values. One area in which ad-
herence to stakeholder values is important, is Agent-Oriented Software Engineer-
ing (AOSE). The Tropos AOSE method, with its concept of soft-goal, comes close
to meeting our principles, but does not address values explicitly. Value-Sensitive
Design is a methodology that does explicitly address value issues, but it offers little
guidance in operationalizing them. We discuss a case study in which we attempt to
capture values in Tropos' soft-goals after eliciting them using Value-Sensitive De-
sign. Subsequently, we discuss to what extent Tropos adheres to our principles.
Finally, we propose the introduction of values as a first-class entity in Tropos in
order to meet our aims of dealing with values.

   Beside eliciting and specifying stakeholder values in the requirements engineer-
ing phase of new technologies, we need to be able to examine how existing tech-
nologies support stakeholders' values.

Research questions How do designer values and stakeholder values affect stake-
holders' acceptance of Pervasive Computing applications for personal well-
being? Do similarities and differences between designer and stakeholder val-
ues impact how stakeholders experience the technology?

In Chapter 7 we investigate the role of user personality and values in relation to
factors influencing the use of Pervasive Computing applications for healthy food
consumption.This research is motivated by the observation that these applications
employ diverse strategies, which are more or less prescriptive, to support healthy
eating behaviors. The choice of strategy reflects the designer's stance on how
behavior change can be achieved. Judgments on values such as autonomy underlie
these stances and we argue, that an alignment of user values and personality and
designer stances will lead to more successful app use and, ultimately, sustainable
health behaviors.

   Figure 1.1 presents an overview of the work in this dissertation.

1
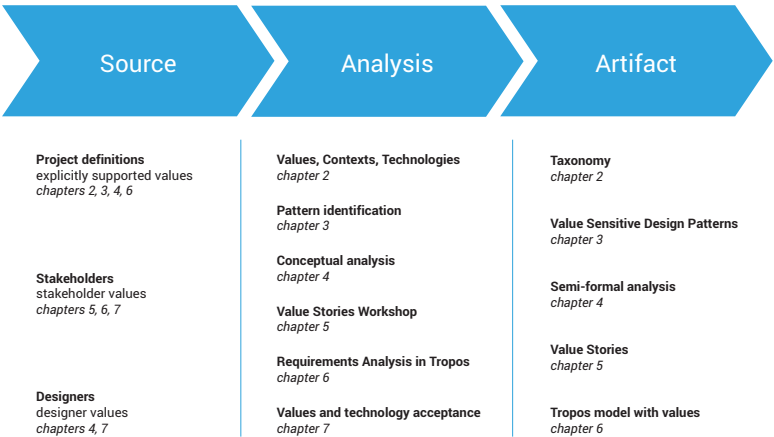
Figure 1.1: Process

1.6. List of Publications

15. **Detweiler, C. A.** and Hindriks, K. V. , *A Survey of Values, Technologies and Contexts in Pervasive Healthcare*, Pervasive and Mobile Computing, Volume 27, Pages 1-13, (April 2016).

14. Zevenbergen, B. , Mittelstadt, B. , Veliz, C. , **Detweiler, C. A.** , Cath, C., Savulescu, J. and Whittaker, M. , *Philosophy meets internet engineering: Ethics in networked systems research. (GTC workshop outcomes paper).*, Social Science Research Network Working Paper Series, (2015)..

13. Harbers, M., **Detweiler, C. A.**, and Neerincx, M. A., *Embedding Stakeholder Values in the Requirements Engineering Process*, Requirements Engineering: Foundations for Sofware Quality (2015).

12. **Detweiler, C. A.** , Harbers, M. and Hindriks, K. V. , *Value Stories: Putting Values into Requirements Engineering*, Proceedings of the Fourth International Workshop on Creativity in Requirements Engineering at REFSQ (2014).

11. **Detweiler, C. A.** and Hindriks, K. V. , *Value-sensitive design patterns for pervasive health care*, Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012 IEEE International Conference on Pervasive Computing (2012).

10. **Detweiler, C. A.** , Dechesne, F. , Hindriks, K. V. and Jonker, C. M. , *Ambient Intelligence Implies Responsibility*, Ambient Intelligence and Smart Environments **12**, (2012).

9. Pommeranz, A. , **Detweiler, C. A.** Wiggers, P. and Jonker, C. M. , *Elicitation of situated values: need for tools to help stakeholders and designers to reflect and communicate*, Ethics and Information Technology **14**, 4 (2012).

1

8. **Detweiler, C. A.** and Pommeranz, A. , *Go slow and beware! A call for reflection on our computational surroundings* , Slow Technology DIS 2012 Workshop (2012).

7. **Detweiler, C. A.** , Pommeranz, A. and Stark, C. L. , *Methods to account for values in human-centered computing* , CHI'12 Extended Abstracts on Human Factors in Computing Systems (2012).

6. O'Kane, A. , **Detweiler, C. A.** and Pommeranz, A. , *Designing and evaluating for trust: a perspective from the new practitioners*, 1st International Workshop on Values in Design–Building Bridges between RE, HCI and Ethics, (2011).

5. **Detweiler, C. A.** , Pommeranz, A. , van den Hoven, M. J. and Nissenbaum, H. , *Values in design-building bridges between RE, HCI and ethics*, Human-Computer Interaction–INTERACT 2011 (2011).

4. Pommeranz, A. , **Detweiler, C. A.** Wiggers, P. and Jonker, C. M. , *Self-reflection on personal values to support value-sensitive design*, Proceedings of the 25th BCS Conference on Human-Computer Interaction (2011).

3. **Detweiler, C. A.** , Hindriks, K. V. and Jonker, C. M. , *Principles for value-sensitive agent-oriented software engineering*, Agent-Oriented Software Engineering XI (2011).

2. **Detweiler, C. A.** , Hindriks, K. V. and Jonker, C. M. , *A Value-Sensitive Approach to Agent-Oriented Software Engineering*, Workshop 8 The Eleventh International Workshop on Agent Oriented Software Engineering AOSE 2010 (2010).

1. **Detweiler, C. A.** and Broekens, J., *Trust in Online Technology: Towards Practical Guidelines Based on Experimentally Verified Theory*, Human-Computer Interaction. Ambient, Ubiquitous and Intelligent Interaction **5612** (2009).

References

[1] I. R. Bartky, *The adoption of standard time,* Technology and Culture **30**, 25 (1989).

[2] D. Kravets, *NSA Wrongly Says Warrantless Mobile-Phone Location Tracking Is Legal,* (2013).

[3] M. Franssen, G.-J. Lokhorst, and I. van de Poel, *Philosophy of technology,* in *The Stanford Encyclopedia of Philosophy*, edited by E. N. Zalta (2013) winter 2013 ed.

[4] T. Bynum, *Computer and information ethics,* in *The Stanford Encyclopedia of Philosophy*, edited by E. N. Zalta (2011) spring 2011 ed.

[5] B. Friedman and P. H. Kahn, Jr., *Human values, ethics, and design,* in *The human-computer interaction handbook*, edited by J. A. Jacko and A. Sears (L. Erlbaum Associates Inc., Hillsdale, NJ, USA, 2003) Chap. 61, pp. 1177–1201.

[6] T. W. Bynum, *Norbert wiener and the rise of information ethics,* in *Information Technology and Moral Philosophy*, edited by M. J. van den Hoven and J. Weckert (Cambridge: Cambridge University Press., 2008) pp. 8–25.

[7] B. Friedman, P. Kahn Jr, and A. Borning, *Value sensitive design and information systems,* Human-computer interaction in management information systems: Foundations **4** (2006).

[8] A. Borning, B. Friedman, J. Davis, and P. Lin, *Informing public deliberation: Value sensitive design of indicators for a large-scale urban simulation,* in *ECSCW 2005*, edited by H. Gellersen, K. Schmidt, M. Beaudouin-Lafon, and W. Mackay (Springer Netherlands, 2005) pp. 449–468, 10.1007/1-4020-4023-7_23.

[9] C. A. Le Dantec, E. S. Poole, and S. P. Wyche, *Values as lived experience: evolving value sensitive design in support of value discovery,* in *Proceedings of the 27th international conference on Human factors in computing systems*, CHI '09 (ACM, New York, NY, USA, 2009) pp. 1141–1150.

[10] A. Borning and M. Muller, *Next steps for value sensitive design,* in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12 (ACM, New York, NY, USA, 2012) pp. 1125–1134.

[11] B. Friedman, *Value sensitive design,* in *Liberating voices: A pattern language for communication revolution*, edited by D. Schuler (The MIT Press, 2008) pp. 366–368.

1

2

A Survey of Values, Technologies and Contexts in Pervasive Healthcare

[1]

We argue that pervasive computing technologies for elderly care can have beneficial and harmful ethical implications. At the heart of these ethical implications lie the effects technologies have on human values, such as well-being, autonomy and privacy. A technology's functions influence how if affects values. These functions are the result of design decisions. So, design can play a part in dealing with ethical implications.We argue that by understanding the relationship between values and technologies in this domain, designers will be in a better position to account for values explicitly, and hence address ethical implications, throughout design. To foster such an understanding, we survey literature on pervasive computing for elderly care, and identify values, technologies and contexts discussed there. We develop a taxonomy to categorize our findings, which serves as a basis to identify and analyze relationships between values, technologies and contexts in pervasive computing for elderly care. With this analysis, we aim to help designers consider the ethical implications of their designs.

2.1. Introduction

It is not unusual for an article on pervasive computing to begin by asserting that pervasive computing technologies hold the promise to help older adults maintain their independence, by supporting them in their daily activities, improving their safety and quality of life, while enabling them to remain living comfortably in their own home as long as possible, "aging in place". In living up to this promise, pervasive computing is likely to contribute to the good of many older adults, and benefit society at large by addressing some of the challenges of an aging population. In other words, these technologies have moral or ethical implications; their contribution to the well-being of older adults is good, and helping older adults maintain their independence is the right thing to do. Pervasive computing technologies can also have less desirable ethical implications, such as privacy concerns and loss of autonomy [1].

Such ethical implications center on pervasive computing technologies' role in supporting or hindering human values. For example, the broad aims of many pervasive computing technologies for elderly care are based on human values, such as independence, safety and well-being. Some of pervasive computing's negative ethical implications have to do with human values, such as privacy and autonomy. Values can be defined broadly as "what a person or group of people consider important in life" [2]. Examples of values include human welfare, fairness, justice, privacy, accountability, and autonomy. It is when a technology affects values favorably (for example, improves quality of life) or adversely (for example, violates privacy) that ethical issues arise. Technologies affect values by providing or performing functions that make possible or support some actions, and not others. A fall detection system might enhance (positively affect) an older adult's safety by notifying caregivers of falls. The same fall detection system, if it captures images of the older adult, might diminish (negatively affect) that person's privacy.

Though a technology's effects on values occur in the context of use, they begin to take shape earlier. During design, designers make decisions on which functions a technology will include, and which it will not. Designers also decide how the technology will provide these functions (for example, fall detection using an accelerometer instead of a camera). Since these functions play an important role in how a technology affects various values, and designers make fundamental decisions about these functions, designers arguably play a part in shaping how a technology will affect various values, with potential ethical implications. In some cases, they do so explicitly, for example, aiming to help older adults maintain their independence. In other cases, the effect the technology has might not be apparent until it is put to use. Either way, designers play a part.

Considering that design decisions influence pervasive computing technologies' effects on a range human values, and thus their ethical implications, relevant values should be taken into account during design. To make values a design consideration, a designer needs to have a definition of values, some awareness of values already at play in pervasive computing and elderly care, a way to identify and analyze yet unaddressed values in this field and tensions among them, and an understanding of the mechanisms through which various technological features affect values. Val-

ues, and their relationship with technology, are not necessarily familiar concepts to designers in this field. Nor are ways of dealing with them.

Value Sensitive Design (VSD) is an established approach for addressing human values throughout a technical design and implementation process [2]. It offers a range of techniques to identify and analyze a technology's stakeholders and their values. It has been applied in a range of domains, including ubiquitous computing [3]. To our knowledge, it has not been applied in the area of pervasive computing for elderly care. Furthermore, VSD has not been used to examine the relationships between values and the functions provided by pervasive computing technologies for elderly care. Values, such as privacy and trust, do receive a fair amount of attention in research on pervasive computing, but they are often discussed in individually and within the scope of a specific project (for example, privacy concerns that arose in evaluating a fall detection system). To gain an understanding of the many values at play in this domain, the relationships between them, and the ways in which various pervasive computing technologies affect them, one would have to assimilate discussions of values that are spread out in research papers across this field.

We aim to provide an overview of the values at play, and relationships between these values and the technologies being developed in pervasive computing for elderly care. We argue that by understanding these relationships, designers will be in a better position to account for values explicitly throughout design. We develop this overview by identifying values, technologies and their contexts in this field.Using concepts and techniques from VSD, we survey literature on pervasive computing for elderly care, identifying stakeholders, explicitly supported values, and the ways in which these are affected. We identify the technologies that contribute to applications in this field, and the contexts in which they are used. We present a taxonomy that consists of different categories of values, technologies and contexts, which we used to classify the entities we identify. We use this taxonomy as a basis for identifying and analyzing relationships between values, technologies and contexts in pervasive computing for elderly care.

This paper is organized as follows. In Section 2.2, we describe our approach. We introduce Value Sensitive Design and describe how it can be used as an analytical tool. We also discuss a literature survey we conducted as a starting point for our analysis, as well as the taxonomy we developed to support our analysis. In Section 2.3, we present the results of our literature survey and classification in our taxonomy. In Section 2.4, we discuss our analysis of relationships between values, technologies and contexts in pervasive computing for elderly care. We present our conclusions and suggestions for future work in Section 2.5.

2.2. Approach
2.2.1. Value Sensitive Design

We use a number of key concepts from the Value Sensitive Design (VSD) framework to help identify values, technologies and contexts of use in pervasive computing for health care, and relationships among them. VSD is an approach to technology design that aims to account for human values systematically throughout the design

process [4]. VSD uses conceptual, empirical and technical investigations to discover and define values, and analyze how existing technologies affect these values, or design new technologies to support values. Throughout these investigations, VSD uses a number of key concepts that our relevant to our present work, namely stakeholders, values and potential benefits and harms.

VSD defines values broadly as "what a person or group of people consider important in life" [2]. Work on VSD often focuses on moral values, such as human welfare, trust, privacy and accountability [2]. VSD takes an interactional stance on values, which holds that values are neither solely determined by technology nor by social systems in which technology is deployed and used, but through the interaction of both [5]. That is, technological development is influenced by people and the social systems they are part of, and technologies contribute to shaping individual behavior and social systems. VSD distinguishes between explicitly supported, stakeholder and designer values [6]. Explicitly supported values are those a technology aims to support. For example, a fall detection system explicitly aims to enhance its users' safety. Stakeholder values are those held by the stakeholders of a technology. A technology can affect these values, though they are not necessarily considered (or even known) during design. For example, users of a fall detection system that uses cameras might feel that the system invades their privacy. Finally, designer values are those a designer holds, which are not necessarily expressed, and often guide designers' decision making subtly, without their awareness. For example, someone designing a fall detection algorithm might choose to favor false positives (the user does not fall, but the algorithm detects a fall) over false negatives (the person falls, but the algorithm does not detect a fall), on the implicit grounds that "it is better to be safe than sorry". This decision involves a value-judgment, rather than a factual one (cf. [7]), and is guided by the designers' values.

Stakeholders are another key VSD concept that we use here. VSD distinguishes between direct and indirect stakeholders. Direct stakeholders are the individuals who interact directly with the technology or the technology's output. Indirect stakeholders are those who do not interact the technology (or its output) directly, but are affected by it. The latter group is often overlooked in the design of technology, which understandably focuses on direct users.

VSD identifies and accounts for these concepts using conceptual, empirical and technical investigations iteratively and integratively. Conceptual investigations focus on identifying which direct and indirect stakeholders are affected by a technology and what values are implicated. In these investigations, the designer can identify benefits and harms that the technology potentially poses to its stakeholders. Designers can use these harms and benefits to identify underlying values, and tensions between the values. Subsequently, designers can use relevant literature to define the identified values. Conceptual investigations are informed by empirical investigations of the technology's context. Empirical investigations can take many forms as "the entire range of quantitative and qualitative methods used in social science research is potentially applicable" [2]. Finally, technical investigations assess how existing technologies support or hinder human values. Technologies provide certain "value suitabilities" [2], which technical investigations examine. Alterna-

tively, technical investigations can consist of designing a system to support selected human values.

We use a combination of conceptual and technical investigations to identify direct and indirect stakeholders and values in pervasive computing for health care, and examine how technologies in this field affect those values. Our focus here is on explicitly supported values, as the main source of our analysis is pervasive computing literature.

2.2.2. Literature Survey

We need sources from which to identify values, technologies and contexts in pervasive computing for health care. We used academic literature on pervasive computing as a starting point, as it is a useful way of getting an overview of the state of the art in this field. Moreover, literature describes technologies at a level of detail that is conducive to our analysis. We selected academic journals that focus on pervasive computing, or related research topics including ubiquitous computing, ambient intelligence, smart environments and internet of things, as well as human-computer interaction. Our selection included Personal and Ubiquitous Computing; Pervasive and Mobile Computing; IEEE Pervasive Computing; International Journal of Ubiquitous Computing; Interacting with Computers; Human-Computer Interaction; and International Journal of Human-Computer Studies.

We searched these journals' archives to find articles that fit our focus, using the following keywords: *independent living*; *(ambient) assisted living*; *aging* (or *ageing*) *in place*, *at home*, or *gracefully*; *elderly care*; *eldercare*; *health care for elderly*; *assistive technology*; *supporting activities of daily life*. To ensure that articles in the human-computer interaction journals we selected fit the overall area of pervasive computing and related research topics, we searched using keywords *ubiquitous computing*; *pervasive computing*; *internet of things*; *physical computing*; *smart homes*; *ambient intelligence*; *intelligent environments*; *smart environments*; *post-desktop computing*; *embedded computing*; and *sensor networks*. We opted to limit our selection to articles published from 2000 onwards, because, with the exception of Personal and Ubiquitous Computing (known as Personal Technologies from 1997 to 2000), the journals in our selection that focus on Pervasive Computing and related topics explicitly did not emerge until the 2000s. Furthermore, the human-computer interaction journals in our selection each feature two or fewer articles on pervasive computing or related topics prior to 2000.

We refined the selection of articles by scanning their abstracts for terms related to the application domain, elderly care (which we listed in the previous paragraph) and for mention of an application or technology. We scanned the remaining articles for mention of human values, using the definition we presented in Section 2.2.1 as a guide. We also used a heuristic list of values often implicated in system design presented in [2], which consists of: human welfare, ownership and property, privacy, freedom from bias, universal usability, trust, autonomy, informed consent, accountability, courtesy, identity, calmness, and environmental sustainability. This list provides definitions of those values, as well as references to examples of relevant literature, which aided the selection process. This yielded a group of papers

on pervasive computing and related research topics that presented a technology in the elderly care domain, and discussed one or more human values. We used this selection as a source for our analysis.

2.2.3. Taxonomy

The process we described in Section 2.2.2 yielded a selection of articles that serves as a source to analyze relationships among values, technologies and contexts in pervasive computing for elderly care. The first step in this analysis is to identify individual values, technologies and contexts described in each of the articles. We group these individual entities in a taxonomy of values, technologies and contexts, based on common characteristics. Using these groups as a starting point, we analyze recurrent relationships between values, technologies and contexts in pervasive computing for elderly care.

To identify values in the articles we selected, we used the definition of values given in Section 2.2.1 and used the heuristic list of values (and definitions of those values) described in Section 2.2.2 as starting points. Table 2.1 lists the values we initially identified. The next step was to group similar values to create categories of values. To guide this process, we turned to Cheng and Fleischmann's meta-inventory of values, which builds on a review of research to date on values in a range of fields including psychology, sociology, anthropology, science and technology studies, and information science, and a review of 12 prominent value inventories developed within those fields [8]. The meta-inventory consists of 16 categories of values: freedom, helpfulness, accomplishment, honesty, self-respect, broad-mindedness, creativity, equality, intelligence, responsibility, social order, wealth, competence, justice, security and spirituality. Further categories that occurred in fewer than 5 of the inventories they reviewed include health, social relationship, trust, a comfortable life, and privacy.

Based on this, we created the categories of values listed in Table 2.2. The values grouped under *physical well-being* correspond roughly to the meta-inventory category *helpfulness*, as it includes the values *human welfare* (which is defined in terms of physical well-being in the heuristic list presented in Section 2.2.1) and *the good life*, and to the *health* category. We rephrased the meta-inventory category of *social relationship* as *social well-being*, to cover the wider range of related concepts we encountered. The *safety* category corresponds roughly to the meta-inventory category *security*, but we rephrased it to avoid confusion with information security, which is an often-addressed topic in pervasive computing. The *freedom*, *privacy*, and *responsibility* categories correspond to the meta-inventory categories of the same names. The remaining values we encountered (self-efficacy, remembering, trustworthiness, and transparency) are not sufficiently similar to any of aforementioned categories to warrant inclusion in them. As these values are each mentioned in fewer than three of the papers included in our survey, we did not create categories for them.

We take a similar approach to identify and group technologies, which is the second dimension of our taxonomy. First, we examine the selection of literature for mention of technologies, including hardware, software, applications and/or archi-

Table 2.1: Initial list of values

Safety	Health	Comfort	Well-being	Independence
Quality of Life	Autonomy	Privacy	Informed consent	Reduced isolation
Freedom	Dignity	Duty of care	Responsibility	Sociability
Mobility	Social stigma	Anonymity	Acceptability	Awareness
Self-efficacy	Assistance	Social Inclusion	Trustworthiness	Transparency
Peace of mind	Remembering	Social contact	Security	Social acceptability
Social welfare	Accountability			

Table 2.2: Value categories

Value category	Identified values
Physical well-being	quality of life, health, comfort
Social well-being	Social welfare, social stigma, social inclusion, social contact, sociability, reduced isolation, social acceptability
Freedom	freedom, mobility, autonomy, independence
Privacy	privacy, informed consent, anonymity
Responsibility	accountability, duty of care, responsibility
Safety	safety, security

tectures. We use Cook and colleagues' [9] survey and classification of the technologies that comprise ambient intelligence to guide the categorization of the technologies we encountered. Their classification is useful for our aims here, as it covers both hardware and software at a level of technical detail that can help in analyzing how technologies affect values in this domain. Cook and colleagues identify five key technologies that contribute to ambient intelligence: sensing, reasoning, acting, security and HCI. They identify a number of types of reasoning, namely user modeling, activity prediction and recognition, decision making, and spatiotemporal reasoning, which we treat separately here. Cook and colleagues' HCI category consists of context awareness and natural interfaces. We include graphical user interfaces (GUIs) in this category, as eight of the articles in our survey discuss pervasive computing applications that include a GUI. We do not discuss the natural interfaces category, as only two of the articles in our survey discussed interfaces that fell within this category, and did so too briefly to contribute to our analysis. We do not discuss the acting or security categories, as we encountered fewer than three papers that presented technologies that fell within each of these categories.

The third dimension of our taxonomy is contexts of use, which we include because values can be defined differently in different contexts, and technologies can have different applications in different contexts. The view of context of use we take in our taxonomy draws on the components of context of use analysis presented in[10], which consist broadly of stakeholders, users, tasks, and environments. In line with the view of stakeholders we presented in Section 2.2.1, we consider users to be direct stakeholders of a system. Accordingly, the context of use dimension of our taxonomy consists of stakeholders, goals or tasks, and environment. This

dimension allows us to categorize the technologies we surveyed according to who uses them or is affected by them, which activities the technologies support, and the social and physical environment in which the technologies are embedded.

2.3. Taxonomy

2.3.1. Values

Physical well-being

One aspect of physical well-being that is frequently mentioned in the literature we surveyed is quality of life. Quality of life is presented as a condition of elders that can be enhanced, improved or increased [11–21], or hampered, diminished or decreased [13, 20, 22, 23]. Improvement is considered to come from applications and services, where as decreased quality of life comes from health risks (e.g., falls [22] or obesity [20]), difficulties caused by dementia [23], blood pressure monitoring [16]. Though the term is used frequently in the literature we surveyed, we did not encounter any definitions.

Comfort is similarly presented as an aspect of elders' lives that can be enhanced [24], improved [21], ensured [15], and felt [16, 18]. It can also be jeopardized [24]. In the latter case, living in one's own home is identified as a potential cause. In the other cases, comfort can result from services that technologies provide, or directly from hardware (in cases where it is comfortable or uncomfortable to wear). As with quality of life, we did not encounter definitions of comfort.

We also included health in the physical well-being category. Health is described as something that can be monitored [11, 16, 18, 20, 21, 24], improved, aided or benefited [25, 26], at risk or hazard [19, 22, 27], has a status or state[1, 16, 21, 28, 29], can have problems [28], can decline [15, 21, 30] and can have complications [31]. We did not find explicit definitions of health in the literature.

Social well-being

In the literature we surveyed, social well-being is often discussed in terms of social contact or connection, social bonding and socializing, and is valued by older adults or their relatives or caregivers [1, 13, 14, 18, 25, 26, 28]. Social contact or connection is described as something that can be facilitated ([14, 18]), stimulated or promoted ([28], and created ([26]). It is discussed as a connection between elders and their (remote) relatives ([1, 14, 18, 28]). Social well-being is adversely affected by social isolation, which can stem from difficulties related to cognitive impairments or dementia ([13, 23]), old age ([18, 26]), or living away from offspring ([20]). Social isolation, and related loneliness, could be reduced by fostering social connection ([26]). Another aspect of social well-being, related to use of technology, is social stigma technology brings ([1, 13]) or, conversely, technology's social acceptability ([30]).

Freedom

The freedom category of the values dimension of our taxonomy includes the value of independence. Independence receives a considerable amount of attention in pervasive computing for elder care, as many of the technologies in this domain

focus on helping elders live independently. Accordingly, it is seen to be a value held by older adults. Though independence is not explicitly defined in the literature we surveyed, much of its meaning can be deduced from the ways in which the term is used. Independence is frequently presented as an aspect or way of living one's life (e.g., "living independently") [1, 11, 13, 15, 17, 19, 20, 22, 23, 28, 32]. Independence is often discussed in conjunction with the idea of living in one's own home ([1, 14, 14, 15, 20, 21, 23, 24, 32]), which suggests this is considered an important aspect of independence. Independence can also refer to independence from caregivers or physician visits [20]. Performing basic activities of everyday living (e.g., bathing and eating) and instrumental activities of daily living (e.g., cooking healthy meals and doing laundry) on one's own (i.e., independently) is a precondition of independence in everyday life [1, 21, 25, 33].

The literature we surveyed mentions a variety of ways in which independence can be affected. It can be facilitated, supported or assisted ([18–20, 23, 25]), maintained ([1, 11, 26, 34]), increased or improved ([12–15, 28, 32]), and promoted ([33]). Independence can be hindered by (the consequences of) falls ([19, 22]), cognitive impairments or dementia ([13, 14, 28, 33, 34]), long-term blood pressure monitoring ([16]), and diseases and disabilities associated with aging ([17, 20, 26]). Notably, [1] mentions that overreliance on technology causes one to disuse one's competences, which can adversely affect independence. Tensions can exist between independence and safety, in that living independently can pose challenges to safety [15]; and between independence and privacy, when technologies to support independent living adversely affect privacy [1].

Autonomy is a similar concept, and is often used interchangeably with independence. Unlike independence, we did encounter a definition of autonomy in the literature: the right to self-determination [18]. Like independence, autonomy has the precondition that one can perform (basic and instrumental) activities of daily living on one's own ([1, 21, 33]), or can self-manage one's life ([20]). Autonomy can be increased, improved or enhanced ([12, 14, 17, 20, 33]). It can be promoted by aging-in-place [21]. Autonomy can also be lost when one is (physically) dependent on technology or receives too much support [1]. One's autonomy can be violated if one's activities are curtailed, or if one perceives or fears curtailment (and thus does not engage in those activities) [18].

Privacy

Privacy is a widely recognized value in this domain. The literature we surveyed discusses privacy exclusively as a value held by older adults. Privacy is described as "a socially constructed value that differs significantly across environments and age cohorts of individuals" [18, p. 2]. In the area of home-based computing, privacy has been defined as seclusion (the right to be left alone); autonomy (the right to self-determination, which is violated through chilling effects when people do not engage in certain activities for fear of surveillance); property (the right to determine the uses and dissemination of personal information); a spatial construct or location privacy (establishing and respecting boundaries); and data protection (generated data are subject to consent to use and correction by the individual) [18]. Discussions of privacy in the literature we surveyed suggest that it can be preserved,

enhanced, ensured, respected or maintained [12, 15, 19, 27, 35, 36]. It can also be given up, sacrificed, compromised, invaded, or impinged on [1, 12, 18, 19, 22]. Furthermore, the literature mentions privacy constraints ([12]), privacy concerns ([1, 13, 30]), privacy issues ([29, 32, 36]), privacy implications ([1]), and privacy related conflicts ([20]). In addition, privacy constraints ([12]), privacy levels ([35]), and privacy requirements ([29]) are mentioned.

Giving up or sacrificing one's privacy is associated with subjecting oneself to observation [12]. More specifically, people's privacy is said to be compromised by capturing images of them [19, 22]. Accordingly, privacy concerns or issues are raised by exposing one's identity [13, 15] and by visual and auditory sensing [1, 30, 32, 36].

Privacy is said to be preserved, maintained or enhanced by having full control of the type of information that different people can access, only disclosing the right information to the right people [12, 35], and by not being able to identify people from information ([12, 13, 15, 27, 29, 32, 36]). Another way of avoiding conflicts is getting people's informed consent ([20]).

It is argued that usefulness of a technology with respect to individuals' own lives plays a role in determining how they think about privacy with regards to that technology — the more useful it is, the more willing individuals are to accept privacy implications [1, 18]. Data granularity or level of intrusion, along with particular reasons for collecting information, have also been found to play a role in individuals' acceptance of privacy implications [1, 18]. Furthermore, activity sensitivity (i.e., potentially sensitive behaviors such as sex) has been found to be an important factor, more so than spatial privacy [18]. It is also suggested that data recipients, that is, who can access sensitive information, are an important aspect of privacy [1, 12, 18, 29, 31, 35].

Responsibility

In the literature we surveyed, the concept of responsibility was most commonly used to convey that a certain hardware (e.g., a mobile device or a sensor) or software component (e.g., a software agent, an application, or a server) has a certain role to perform (e.g., collecting data) [11, 12, 14, 16, 20, 29, 32, 35]. In some cases, it referred an individual's role or duty to do something (e.g., a caregiver's duty to do something) ([23, 27, 29, 34]), or to assume a duty (take responsibility for caring for one's own health) [28]. The term tasks was used to convey the same notion with regards to software or hardware components [11, 12, 24, 32]. With regards to individuals, the concept of task was generally used to refer to (household) activities to be done or performed by an individual as part of everyday life (e.g., washing hands) [1, 14, 17, 20, 23, 26, 27, 33, 34], or as part of a job or role (e.g., caring tasks) [35]. The term "duty" was used in the same sense, exclusively with regards to caregivers [14, 27].

Safety

In the literature we surveyed, safety (occasionally referred to as security) is most frequently discussed as a value held by older adults, (e.g., "the safety of elders") [1, 12, 18–21, 24, 32, 36, 37] or people with dementia [13, 14, 27, 34], and as

a value of caregivers or relatives of older adults (e.g., "family members' desire to keep elders safe") [1, 18]. The term "safe" is also used to describe a living condition (e.g., "live safely") [15, 17], and as a condition of an individual's environment or context (e.g., "a safe living environment") [15, 20, 32].

One can experience feelings of safety or a sense of security [14, 19, 20]. Safety (or feelings thereof) can be enhanced [17, 18, 24, 36], improved [12, 21], increased [20, 32], guaranteed [24], and ensured [1, 15, 27]. On the other hand, safety can be jeopardized [24] or abrogated [18]. Also, one can stay safe [12] or keep (others) safe [1].

The literature suggests that safety is hindered by (health) risks, harm, hazards, hazardous situations, danger, dangerous events or situations, emergencies or emergency situations, being trouble, and experiencing problems [12, 14, 15, 18, 19, 24, 27, 32, 34, 36, 37]. These challenges to safety can stem from a variety of issues, including falls and related injuries [1, 12, 18, 19, 21, 24, 32, 36], unattended events and incomplete activities [15, 32, 37], navigation problems such as wandering [13, 14, 27].

The literature mentions relationships between safety and other values. [15] uses the term "independent but safe living", which suggests that independent living can challenge safety. It has also been argued that individuals might be willing to accept some invasion of their privacy for safety reasons [12, 18], suggesting that in some situations safety outweighs privacy.

2.3.2. Technologies

Sensing

Generally speaking, sensors enable pervasive computing applications to gather sensory data from the physical environment in which they are situated. Anonymous, binary ('on' or 'off') sensors attached to household objects or infrastructure, such as movement detectors, door sensors, contact switch sensors and pressure pads can be used to detect or locate a person's presence in a room or on furniture, or to detect people's interactions with household objects (e.g., a water kettle) as they perform activities [12, 14, 15, 17, 21, 23, 32, 37]. [30] uses user-worn magnetic sensors to detect magnetic fields emitted by electronic devices with which the user interacts.

Body-worn accelerometers can provide information on a person's body movement, which can be used to detect falls or assess levels of physical activity [11, 22, 24, 26, 32]. Imaging sensors (including thermal imaging [19] and 3D sensors[36]) can also be used to gather information on body movement such as posture, position and velocity, which is useful in fall detection applications [1, 18, 24, 32].

Sensors for temperature, blood pressure, pulse, blood-oxygen ratios, ECG or glucometers can be used to gather an individual's vital statistics, for use in health monitoring applications [16, 20, 29, 31]. Temperature, noise and light sensors can provide information on the state of the user's environment, which can help applications that take the user's context or situation into account [17, 29].

NFC/RFID tags and QR codes can be used to uniquely identify and locate tagged objects (e.g., a teacup's presence on a tray), or to store (a link to) relevant informa-

tion (e.g., medication instructions) [28, 33, 35]. Similarly, Bluetooth or modulated illumination-based beacons deployed throughout the user's environment can be used to transmit unique location identification codes, which a hand-held device or user-worn "badge" can detect in order to locate the user [13, 27]. GPS can also be used to determine location [11, 12].

Modeling

Many pervasive computing applications for health care use models of user behavior to adapt the software to the user, and as a baseline from which to detect anomalies or changes in the individual's behavior patterns. We encountered a distinction between models that are specified (or predetermined) based on what we can loosely call "domain knowledge", and those that are built from collected sensor data. The most basic form of the former sets thresholds that specify which input data correspond to "normal" behavior or situations, and which correspond to "abnormal" behavior or situations (e.g., an excessive period of inactivity after a fall) [19, 22, 24, 26, 36]. Similarly, rules can specify actions to take under certain conditions (e.g., sensor readings indicative of arrhythmia) or at specific times [11, 14, 20, 27]. Other applications use ontologies to model user activities as primitive actions or behaviors and associated contexts, and specify which sensors indicate user interaction with those contexts [15, 17, 23, 32].

Another group of applications (automatically) builds or learns activity models from data sources including room occupancy statistics gathered from sensors [12, 21], features extracted from sensor data [30], and descriptions of activities and relevant objects mined from the Web [37].

Activity prediction and recognition

A number of applications in our survey use different types of sensor information, activity models and reasoning to infer what the user is doing. Several applications take a rule-based approach to recognize activities of daily living or events from sensor information. Predefined rules or sequences of rules specify which sensors, sensor values or states (e.g., kettle sensor is on), and, in some cases, time frame and location correspond to a specific activity (e.g., making a cup of tea) or event (e.g., a fall) [12, 23, 32, 36]. A similar approach uses sensor readings, and a predefined ontology that specifies hierarchical relationships between sensors, (household) objects or contexts, and activities or tasks to infer the current activity or task using evidential operations [15] or case-based reasoning [17].

Other approaches to activity recognition use automatically built activity models. These models focus on users' interaction with objects (e.g., a cup and a kettle) in performing activities (e.g., making a cup of tea), or on users' presence (over time) in the rooms of the house. They use models of the relevance weight of objects in activities [37], or features extracted from hand-worn sensors that sense magnetic fields emitted by electrical devices to discriminate activities [30]. The former infers the user's activity from sensor reports indicating her interaction with key objects related to that activity, while the latter computes the activity class of each feature vector extracted from the hand-worn sensors. An alternative approach uses room occupation statistics obtained from motion sensors to build models of Circadian

Activity Rhythm based on presence, activity levels and time spent performing individual activities, where rooms are taken to be indicative of certain activities (e.g., bedroom indicates sleeping) [21]. Based on these models, it can recognize activities and identify behavioral anomalies (i.e., changes in behavioral patterns).

Decision making

Many technologies we surveyed use decision making and planning techniques from artificial intelligence to take action based on detected events. This is often achieved by checking detected events against specified rules that set conditions under which to trigger actions, such as notifying caregivers [1, 12, 16, 17, 19, 20, 22, 24, 27, 29].

Other approaches involve triggering alerts based on detected and classified deviations from baseline activity patterns [21]; inferring whether a user is stuck in performing a task and prompting the user according to inferred cognitive difficulties [23]; and filtering user information according to privacy concerns before disseminating it to appropriate parties [35].

Spatiotemporal reasoning

Technologies in this category make use of reasoning techniques that explicitly represent elements of space and/or time and relationships between them. In its most basic form, it uses the user's location, and predefined spatial information or rules pertaining to spaces (e.g., physical barriers and physical distance, or alert areas) to provide users with relevant information (e.g., the optimal route in a navigation application, or alerts when a user might be at risk) [13, 27]. Another approach compares sensed spatial features to the (predefined) spatial characteristics of specific events (e.g., falls) to detect those events [19, 22, 24, 36]. Temporal features can also be used to detect that a specific event *has* occurred, based on predefined temporal characteristics of the event having occurred (e.g., excessive inactivity in risk areas indicative of a fall) [19].

Specific activities can be detected by reasoning over sensed spatial features (e.g., user location and objects with which the user interacted) using a (predefined) ontology of relationships between objects, contexts and activities [15]. User context (including activity pattern) can be inferred from a combination of sensed spatial and temporal features of events (e.g., user location, position, object of interaction, and time stamps) by software agents using rules specified using concepts from ontologies (e.g., Smart Home, Event and Context ontologies) [32].

Other approaches use motion events detected by sensors throughout the house to gather statistics on activity or presence levels [12, 21]. These systems can calculate relative mobility levels for spaces and periods of interest, and automatically build models of "normal" activity and behavioral patterns. These approaches reason about presence and activity levels in various spaces at various times to detect deviations.

Context Awareness

Cook and colleagues identify context awareness as a key contributing technology to human interaction with ambient intelligence systems [9]. Context awareness uses devices (e.g., sensors) to infer users' current activity and the characteristics of their

environment to intelligently manage information content and its means of distribution. Our survey revealed a number of approaches to context awareness. Though several technologies make use of some form of context awareness, the focus here is on technologies that use context awareness for human-computer interaction.

Some approaches we encountered focus on providing users with information tailored to the user and her current context. Chang and Wang's system uses user location (obtained from Bluetooth beacons), personal profiles (including information on physical disabilities and user preferences), and information on environmental barriers (e.g., stairs) to compute a personalized route for the user [13]. The system provides users with directions and instructions at each navigational decision point along the route. The Ambient Annotation System (ANS) uses augmented reality to assist people suffering from Alzheimer's disease [34]. The system allows caregivers to associate digital tags or notes with objects of interest, by selecting objects in digital images of the user's home. A user-worn mobile device captures images from the user's environment from which the system identifies tags. If a tag is detected, the user is notified and an image of the annotated object is displayed with an associated text message.

García-Vázquez and colleagues' Remind-Me, GUIDE-Me and CARE-Me ambient information systems to remind, guide and motivate elderly users to medicate, use context awareness to intelligently provide users with appropriate assistance [28]. Agents for each of the systems include an Activity-Aware service, which monitors the environment and queries a knowledge base containing a context-aware representational ontology and rules in order to determine the type of assistance the user requires (e.g., appropriate medication reminders, guidance, or motivation). The Remind-Me ambient information system contains an additional Medication-Compliance Aware agent, which monitors medication compliance with personalized medication prescriptions. Hoey and colleagues present a context-sensitive prompting system, which uses sensor observations (caused by a user's interactions with his or her environment) and behavioral and environmental models of interactions relevant to completing tasks involved in the user's activities of daily living, which analyzes ongoing activities and prompts the user through the task as needed [23]. Ni and colleagues' Task Support System they build using this approach detects environment contexts such as humidity, temperature, luminance, and the user's location to provide adaptive and personalized task assistance, such as environment control (temperature etc), medication reminders, and call forwarding [17].

Muñoz and colleagues ambient assisted living system is based on a multi-agent architecture in which agents analyze data obtained from sensors in the user's home to infer the user's context, using Smart Home, Event and Context ontologies [32]. The system aggregates context information from agents that manage each type of context information. Activity patterns are inferred from this information, along with any absence of movement. The system then evaluates the inferred context as safe or unsafe, and triggers an emergency mechanism if needed.

In Ayala and colleagues' system, agents collect data (from sensors) and monitor this data to determine system context [11]. The system analyzes changes in context and determine if the agent needs to self-adjust (configure or reconfigure

its architecture) given the current situation. If needed, the agent self-adjusts according to a plan, which specifies preconditions for its execution and the actions performed in executing the plan.

Graphical User Interfaces

Several of the technologies we surveyed use graphical user interfaces (GUIs) for a variety of purposes. One common use is monitoring health-related information. The BehaviorScope Web portal allows users to register different types of data sources (motion sensors, camera-based localization sensors, RFID tags, GPS-enabled mobile devices and simple sensors, such as door sensors and temperature or humidity sensors) [12] to share with other users of the system. The system provides different forms of visualization and statistics, and allows users to define email and SMS notifications and alarms. The system's Spatiotemporal Filtering Language GUI also allows users to set triggers for alarms and notifications. The Digital Family Portrait informs older adults' family members of their daily activities, health status, and potential problems, as well as patterns of activities over a time period [1]. Icons for each day visualize the older adult's average activity level by varying in size. Detailed information on each day can be obtained by touching the day's icon. Virone's SAM-CAD II GUI displays plots of presence-based, activity-level-based and activity-based Circadian Activity Rhythms for any day within a selected observation window [21]. It also includes plots of overall behavior deviation (from normal activity patterns), and allows users to plot specific types of alerts. Another panel lets users view behavioral deviations per room or activity. A third panel provides plots of total time spent or activity level per room, or total time spent to perform an activity, per hour of the day over several days.

Other approaches focus on providing users with instructions or reminders, for wayfinding or medication intake, which are tailored to the user's context using information about user location, physical abilities, care needs medication compliance [13, 14, 28, 33]. These are generally presented on handheld devices, as textual instructions and associated images, and video.

2.3.3. Contexts of Use

We examined each of the papers in our selection to identify which stakeholders their technology targeted or would affect, which goals or tasks the technologies were designed to support, and in which environment the technology would be deployed and used. This gave us a list of all of the occurrences of these elements in the papers in our shortlist. We examined the full list of elements of context of use to identify similarities based on which we could create groups of stakeholders, goals, and environments. For example, we found mentions of "older adults", "elders", "elderly people", and "seniors". We take these terms to refer to the same concept, and labeled the group "elders".

We identified the following groups of stakeholders.

- elders

- people with a cognitive impairment

- professional caregivers

- family and friends

We identified the following groups of goals the technologies aimed to support.

- maintain health

- guarantee safety

- control surroundings

- cope with memory loss

- perform activities of daily life

Finally, we created the following groups of environments in which the solutions we surveyed were deployed.

- home

- outdoors

- residence (i.e., assisted living facility)

- workplace

Any combination of three elements (one from each group) is a potential context of use. So, there are 80 possible contexts of use. With 37 papers in our shortlist, it did not seem likely that all of these possible contexts would be targeted in the papers we surveyed. To check this, we reexamined which stakeholders, goals and environments each shortlisted paper addresses, and categorized the papers according to the groups presented above. We then counted how many papers addressed each potential context of use.

In deciding which contexts of use (i.e., combinations of stakeholders, goals and environments) to include along the contexts of use dimension of our taxonomy, we opted not to include any contexts that were addressed by fewer than three papers. As a result, we excluded 75 of the contexts of use, and were left with five stakeholder-goal-environment combinations:

- elders - maintain health - home (9 papers)

- elders - guarantee safety - home (7 papers)

- professional caregivers - maintain health - home (5 papers)

- elders - maintain health - outdoors (3 papers)

- elders - control surroundings - home (3 papers)

2.4. Synthesis: Values, Contexts of Use, and Technologies

We examined all possible combinations of a value (physical well-being, social well-being, privacy, freedom, responsibility or safety), a technology (sensing, modeling, activity recognition, spatiotemporal reasoning, decision making, acting, context awareness, natural interfaces, graphical user interfaces) and a context (⟨ elders, maintain health, home ⟩ , ⟨ elders, guarantee safety, home ⟩ , ⟨ professional caregivers, maintain health, home ⟩ , ⟨ elders, maintain health, outdoors ⟩ , ⟨ elders, control surroundings, home ⟩), a total of 270 combinations. For each combination, we examined which papers addressed that combination. Of the 270 possible combinations, 137 were not addressed by any papers, 66 were addressed by 1 paper each, 36 were addressed by 2 papers, 14 combinations were addressed by 3 papers, 9 combinations were addressed by 4 papers and 8 combinations were addressed by 5 papers.

Social well-being did not occur in any combinations addressed by 3 or more papers. Acting, context awareness, natural interfaces, and graphical user interfaces did not occur in combinations addressed by 3 or more papers. ⟨ Elders, Maintain health, Outside ⟩ and ⟨ Elders, Control surroundings, Home ⟩ also did not occur in any combinations addressed by 3 or more papers.

By examining how a value, technology and context is addressed by each paper in each combination, we attempt to identify recurring relationships between values, technologies and contexts. More specifically, for every combination, we examine what is deemed important under the value in this context (e.g., under the value of privacy, one's ability to be secluded in one's home could be the valued state of affairs, or the object of value). We identify how the current context supports or hinders that value. We also identify how the technology in this combination (positively or negatively) changes the context, and how the technology can support or hinder the value.

Many aspects of the combinations we identified that were addressed by three or more papers have elements in common. For example, two combinations address freedom and sensing and two combinations address freedom and modeling, in different contexts. As our focus is on values, we discuss the combinations using values as a starting point.

2.4.1. Physical well-being

Sensing

Binary sensors on household objects (e.g., water boilers) and infrastructure (e.g., windows) can contribute to elders' physical well-being by avoiding discomfort of more invasive forms of sensing (e.g., body-worn sensors), and by helping detect health risks and decline, when combined with other technologies (e.g, modeling and activity recognition). Location sensors, in combination with other technologies (e.g., spatiotemporal reasoning), can improve the quality of life of people with dementia and/or cognitive impairments by helping them navigate. User-worn sensors (e.g., accelerometers) can enhance physical well-being, by gathering information that can

improve prevention and treatment, such as activity levels. These sensors can also help detect falls, which are a major health risk. However, user-worn sensors can adversely affect physical well-being if they are uncomfortable. Imaging sensors can improve well-being by helping detect falls, without the discomfort of user-worn sensors.

Modeling

Simple modeling in the form of thresholds and rules for raw or aggregated sensor data can contribute to physical well-being by helping detect specific health decline and risks (hence improving treatment and prevention), while avoiding discomfort of activities such as long-term blood pressure monitoring. Thresholds and rules can also contribute to physical well-being by helping elders maintain healthy levels of physical activity, drawing attention to overly high or low levels. Ontologies can improve quality of life by helping individuals cope with some of the difficulties associated with dementia, and by helping detect hazardous situations that could adversely affect individuals' health. Statistical approaches to (activity) modeling can improve treatment and prevention by providing a model of "normal" activity from which deviations (indicative of health decline) can be detected. Moreover, they avoid the discomfort of more invasive ways of checking individuals' health status.

Decision making

Decision making technologies can improve physical well-being by notifying caregivers of situations or conditions that require their attention and/or assistance, which supports prevention and timely treatment, reducing injury and harm. It can also improve the quality of life of people with dementia, by prompting them when they get stuck performing tasks. It can also help elders sustain healthy levels of activity by prompting them to be more or less physically active.

2.4.2. Freedom

Sensing

Sensors on household objects and infrastructure can enhance elders' independence, allowing them to live in their own home by gathering information (e.g., activity levels) that would normally require caregivers to be present in the elder's home, that the elder live in a nursing home, or that the elder visit a physician. Such sensors can potentially violate elders' autonomy if they prevent elders from engaging in certain activities for fear of surveillance. Location sensors can enhance independence of people facing dementia or cognitive impairments who are otherwise physically mobile, by assisting them in navigating, and by reducing the risks of wandering. These sensors can threaten users' autonomy if they create a chilling effect. User-worn sensors (e.g, accelerometers) can enhance independence by helping to detect risks (e.g., falls), reducing the need for visits from caregivers or placement in a nursing home. They also can help detect signs of health decline, reducing the need for physician visits, which further supports independence. User-worn sensors can violate users' autonomy if they evoke fear of surveillance, and potentially through the burden (and stigma) of wearing sensors. Imaging sensors enhance independence

by helping detect risks such as falls within the home without the burden of wearing sensors. However, the potential of chilling effects, and consequent violation of autonomy, is especially strong with this type of sensor, even if a person is not actually identifiable from the images captured. Sensors that capture vital statistics enhance independence by enabling remote health monitoring, reducing the need for physician visits and hospitalization.

Modeling

Modeling enhances elders' independence by allowing thresholds and rules to be specified react to certain raw sensor input, which helps involve caregivers in elders' lives only when necessary and allows elders to remain in their own homes. Ontologies can enhance independence by helping detect adverse events, thus reducing dependence on caregivers. They can also help elders and people faced with dementia or cognitive impairments perform activities of daily living independently by helping detect (in combination with sensors) when the individual is stuck in an activity and making it possible to provide the necessary information (in combination with decision making, activity recognition and/or spatiotemporal reasoning technology) to get them unstuck. These technologies might adversely affect autonomy if elders perceive overreliance on the technology. Statistical approaches to modeling contribute to remote health status monitoring, and reduce physcan visits and increase independence, by building a representation of "normal" health, enabling detection of anomalies and deviations indicative of health decline.

Spatiotemporal reasoning

Spatiotemporal reasoning technologies can enhance the independence of people facing dementia and cognitive impairments by helping them navigate – providing them with instructions or directions at appropriate points. Also, it can help determine if they are stuck performing an activity of daily living, and provide them with the appropriate information to complete the activity successfully. The same technology could adversely affect these people's autonomy if they believe they are overreliant on the technology. Spatiotemporal reasoning can help detect hazardous situations, reducing the need for placement in nursing homes or hospitalization, thereby increasing independence. These techniques also contribute to remote health monitoring, by helping identify health decline from changes in activity patterns.

Decision making

Decision making technologies can support freedom by checking sensed events against pre-specified rules, and contacting caregivers if their assistance is needed (e.g., in cases of emergencies), increasing elders' independence from caregivers. These technologies can also trigger alerts based on changes in activity patterns that might be indicative of health issues. This can improve treatment, which can prolong elders' stay in their own home, and increase independence. Prompting technologies (which make use of decision making techniques) help people facing dementia or cognitive impairments maintain independence by assisting them in performing activities of daily life autonomously. Decision making techniques can violate older

adults' autonomy when it offers too much support or raises concerns of overreliance on technology.

2.4.3. Privacy

Sensing

Sensors on household objects and infrastructure can support privacy because they do not capture identifiable features of a person directly. They can potentially impinge on spatial privacy and activity sensitivity. Also, they can violate privacy in the sense of seclusion if they are always on and there were no location where the individual could choose to be free from monitoring. These sensors can also violate privacy as autonomy, through fear of surveillance elders might experience. Location sensors can impinge on spatial privacy, activity sensitivity (if certain locations are associated with certain activities), privacy as seclusion, and privacy as autonomy in similar ways. User-worn sensors for body movement can hinder privacy by their potential to capture sensitive activities. Privacy as autonomy can be violated through fear of surveillance of body movements. Imaging sensors add additional concerns of being identifiable from captured data. Such concerns might be alleviated by processing images locally, or only using blob images (from which a person cannot be detected), but only if the monitored person understands the technology. All of these types of sensors might raise privacy concerns related to the granularity of the data being captured, the necessity of the capturing, and who the data recipients are. Also, there are potential concerns with regard to control over data collection and dissemination.

Modeling

Activity models have the potential to negatively affect privacy if their use raises concerns about what alerts are triggered and the ability of elders to correct these alerts (or the underlying models). They also potentially adversely affect privacy if they involve sensitive activities (e.g., time spent in the bedroom). An advantage of using is that they can be responsive to very specific data (e.g., acceleration from a fall) and thus avoid using personally identifiable information, except in cases where statistical models of activity are used (which are essentially unique to an individual). Activity models can also raise concerns related to what they capture for what reason.

Activity recognition

Activity recognition techniques have the potential to negatively affect privacy by detecting sensitive activities. They can also raise concerns over what data is generated at what granularity (i.e., what activity is recognized), to whom this is communicated and for what reason. Also, it may raise concerns over the ability to control what activities are recognized and how to correct information generated.

Spatiotemporal reasoning

Spatiotemporal reasoning techniques can negatively affect privacy by combining information (e.g., times of day spent in certain rooms) that could be indicative of sensitive activities. The effects might be reduced by only comparing sensed

features to representations of specific events (e.g., a long time spent on the floor could be an indication of a fall). Other privacy concerns can include granularity of the spatiotemporal reasoning.

Decision making
Decision making techniques, in the form of notifications to caregivers and other interested parties, can negatively affect privacy by raising concerns about what data is communicated to whom, with what granularity or level of detail.

2.4.4. Safety

Sensing
Sensors on household objects and infrastructure can enhance safety by helping make unattended events and other potentially hazardous situations detectable. Location sensors can protect safety by preventing problems with wayfinding and wandering, helping protect the safety of wandering-prone individuals (e.g., people facing dementia) by tracking there whereabouts. User-worn motion sensors can help detect falls, as can imaging sensors. Sensors for vital signs can help detect health issues.

Modeling
Modeling techniques further improve safety by allowing systems to be responsive to certain sensed events (e.g., falls or excessive inactivity). Ontologies can help by enabling systems to assess whether a situation, environment or context is safe or unsafe, and detect unattended events. Statistical models can enhance safety by enabling early detection of health decline.

Activity recognition
Activity recognition can use models to detect hazardous events, unsafe contexts and health decline in a timely manner.

Spatiotemporal reasoning
Spatiotemporal reasoning techniques enhance safety by helping people facing dementia navigate, and by detecting when they are wandering (which can be hazardous). It can also help detect potentially hazardous situations (e.g., a stove left unattended, or front door left open). Spatiotemporal reasoning can also help identify deviations in activity patterns that might indicate health issues or health decline. These techniques can also be used to detect specific hazardous events (e.g., a person has fallen and has subsequently been inactive for an excessive period of time).

Decision making
Decision making techniques enhance the safety of older adults by notifying caregivers in emergencies, getting elders' the assistance or treatment they need on time, hence reducing harm.

2.5. Conclusion

Pervasive computing has the potential to address many of the problems societies with aging populations face, by making computers available anywhere at any time (or even everywhere and all the time). As computing weaves itself into our everyday lives, the opportunities for it to affect our lives positively and negatively are more numerous than ever before. Pervasive computing's envisioned deep intertwining with our lives makes human values a fundamental consideration for this field.

Designers working in pervasive computing are faced with a wide range of issues to take into account. This increasingly includes issues such as indirect stakeholders, contexts and values, which might not be concepts designers are used to dealing with extensively. Designers are forced to come up with ways of dealing with these unfamiliar concepts. Unfortunately, there are few ways of systematically identifying these issues, and there is little guidance on dealing with these issues in design. Where these issues are successfully addressed in design, the resulting design knowledge is often lost for lack of a systematic way to capture and communicate it. As a result, examples of best practice might not be readily available.

In this paper, we used concepts from VSD to identify values, technologies and contexts of use in this domain. We developed a taxonomy in which we categorized the identified elements. This taxonomy served as a starting point to analyze the relationships between values, technologies and contexts of use in pervasive computing for health care.

We argue that understanding these relationships can help designers better consider the values their technologies affect. We believe this will help designers weave computing technologies into our everyday lives in a way that is sensitive to the needs and values of the people living those lives, and to the contexts in which they are lived.

Our focus in this paper was on explicitly supported values, or at least values that were explicitly mentioned. Explicitly supported values are not the only values to consider. Stakeholders' values and (unarticulated) designers' values should also be considered, in order to get a more complete understanding of which values the technology might (inadvertently) support, hinder, violate or otherwise affect and take this into consideration in design. Future work should develop ways of uncovering these other values as part of analysis.

References

[1] E. Mynatt, A. Melenhorst, A. Fisk, and W. Rogers, *Aware technologies for aging in place: understanding user needs and attitudes,* Pervasive Computing, IEEE **3**, 36 (2004).

[2] B. Friedman, P. Kahn Jr, and A. Borning, *Value sensitive design and information systems,* Human-computer interaction in management information systems: Foundations **4** (2006).

[3] B. Friedman, I. E. Smith, P. Kahn Jr, S. Consolvo, and J. Selawski, *and j.*

selawski. *development of a privacy addendum for open source licenses: Value sensitive design in industry.* Ubicomp , 194 (2006).

[4] B. Friedman, D. Howe, and E. Felten, *Informed consent in the mozilla browser: implementing value-sensitive design,* in *System Sciences, 2002. HICSS. Proceedings of the 35th Annual Hawaii International Conference on* (2002) p. 10 pp.

[5] J. Woelfer and D. Hendry, *Designing ubiquitous information systems for a community of homeless young people: precaution and a way forward,* Personal and Ubiquitous Computing **15**, 565 (2011).

[6] A. Borning, B. Friedman, J. Davis, and P. Lin, *Informing public deliberation: Value sensitive design of indicators for a large-scale urban simulation,* in *ECSCW 2005*, edited by H. Gellersen, K. Schmidt, M. Beaudouin-Lafon, and W. Mackay (Springer Netherlands, 2005) pp. 449–468, 10.1007/1-4020-4023-7_23.

[7] F. Kraemer, K. van Overveld, and M. Peterson, *Is there an ethics of algorithms?* Ethics and information technology **13**, 251 (2011).

[8] A.-S. Cheng and K. R. Fleischmann, *Developing a meta-inventory of human values,* Proceedings of the American Society for Information Science and Technology **47**, 1 (2010).

[9] V. R. J. Diane J. Cook, Juan C. Augusto, *Ambient intelligence: Technologies, applications, and opportunities,* Pervasive and Mobile Computing , 277 (2009).

[10] M. MAGUIRE, *Context of use within usability activities,* International Journal of Human-Computer Studies **55**, 453 (2001).

[11] I. Ayala, M. Amor, and L. Fuentes, *Self-configuring agents for ambient assisted living applications,* Personal and Ubiquitous Computing , 1 (2012).

[12] A. Bamis, D. Lymberopoulos, T. Teixeira, and A. Savvides, *The behaviorscope framework for enabling ambient assisted living,* Personal and Ubiquitous Computing **14**, 473 (2010).

[13] Y. Chang and T. Wang, *Comparing picture and video prompting in autonomous indoor wayfinding for individuals with cognitive impairments,* Personal and Ubiquitous Computing **14**, 737 (2010).

[14] R. Davies, C. Nugent, M. Donnelly, M. Hettinga, F. Meiland, F. Moelaert, M. Mulvenna, J. Bengtsson, D. Craig, and R. Dröes, *A user driven approach to develop a cognitive prosthetic to address the unmet needs of people with mild dementia,* Pervasive and Mobile Computing **5**, 253 (2009).

[15] X. Hong, C. Nugent, M. Mulvenna, S. McClean, B. Scotney, and S. Devlin, *Evidential fusion of sensor data for activity recognition in smart homes,* Pervasive and Mobile Computing **5**, 236 (2009).

2

[16] A. Lorenz and R. Oppermann, *Mobile health monitoring for the elderly: Designing for diversity,* Pervasive and Mobile Computing **5**, 478 (2009).

[17] H. Ni, B. Abdulrazak, D. Zhang, and S. Wu, *Cdtom: a context-driven task-oriented middleware for pervasive homecare environment,* International journal of ubicomp **2**, 34 (2011).

[18] K. Shankar, L. Camp, K. Connelly, and L. Huber, *Aging, privacy, and home-based computing: Designing for privacy,* Pervasive Computing, IEEE **PP**, 1 (2011).

[19] A. Sixsmith and N. Johnson, *A smart sensor to detect the falls of the elderly,* Pervasive Computing, IEEE **3**, 42 (2004).

[20] D. Vergados, *Service personalization for assistive living in a mobile ambient healthcare-networked environment,* Personal and Ubiquitous Computing **14**, 575 (2010).

[21] G. Virone, *Assessing everyday life behavioral rhythms for the older generation,* Pervasive and Mobile Computing **5**, 606 (2009).

[22] J. Dai, X. Bai, Z. Yang, Z. Shen, and D. Xuan, *Mobile phone-based pervasive fall detection,* Personal and ubiquitous computing **14**, 633 (2010).

[23] J. Hoey, T. Plötz, D. Jackson, A. Monk, C. Pham, and P. Olivier, *Rapid specification and automated generation of prompting systems to assist people with dementia,* Pervasive and Mobile Computing **7**, 299 (2011).

[24] M. Aiello and S. Dustdar, *Are our homes ready for services? a domotic infrastructure based on the web service stack,* Pervasive and Mobile Computing **4**, 506 (2008).

[25] V. Meza-Kubo and A. Morán, *Ucsa: a design framework for usable cognitive systems for the worried-well,* Personal and Ubiquitous Computing , 1 (2012).

[26] H. Qian, R. Kuber, A. Sears, and E. Murphy, *Maintaining and modifying pace through tactile and multimodal feedback,* Interacting with Computers **23**, 214 (2011).

[27] D. Taub, S. Leeb, E. Lupton, R. Hinman, J. Zeisel, and S. Blackler, *The escort system: A safety monitor for people living with alzheimer's disease,* Pervasive Computing, IEEE **10**, 68 (2011).

[28] J. García-Vázquez, M. Rodríguez, A. Andrade, and J. Bravo, *Supporting the strategies to improve elders' medication compliance by providing ambient aids,* Personal and Ubiquitous Computing **15**, 389 (2011).

[29] D. Vassis, P. Belsis, C. Skourlas, and G. Pantziou, *Providing advanced remote medical treatment services through pervasive environments,* Personal and Ubiquitous Computing **14**, 563 (2010).

[30] T. Maekawa, Y. Kishino, Y. Sakurai, and T. Suyama, *Activity recognition with hand-worn magnetic sensors,* Personal and Ubiquitous Computing , 1 (2012), 10.1007/s00779-012-0556-8.

[31] A. Jara, M. Zamora, and A. Skarmeta, *An internet of things—based personal device for diabetes therapy management in ambient assisted living (aal),* Personal and Ubiquitous Computing **15**, 431 (2011).

[32] A. Muñoz, J. Augusto, A. Villa, and J. Botía, *Design and evaluation of an ambient assisted living system based on an argumentative multi-agent system,* Personal and Ubiquitous Computing **15**, 377 (2011).

[33] J. Gómez, G. Montoro, P. Haya, X. Alamán, S. Alves, and M. Martínez, *Adaptive manuals as assistive technology to support and train people with acquired brain injury in their daily life activities,* Personal and Ubiquitous Computing , 1 (2012).

[34] E. Quintana and J. Favela, *Augmented reality annotations to assist persons with alzheimers and their caregivers,* Personal and Ubiquitous Computing , 1 (2012).

[35] J. Bravo, D. López-de Ipiña, and R. Hervás, *Rfid breadcrumbs for enhanced care data management and dissemination,* Personal and Ubiquitous Computing , 1 (2012).

[36] R. Planinc and M. Kampel, *Introducing the use of depth data for fall detection,* Personal and Ubiquitous Computing , 1 (2012), 10.1007/s00779-012-0552-z.

[37] P. Palmes, H. Pung, T. Gu, W. Xue, and S. Chen, *Object relevance weight pattern mining for activity recognition and segmentation,* Pervasive and Mobile Computing **6**, 43 (2010).

2

3

Value Sensitive Design Patterns for Pervasive Healthcare Computing

Pervasive health computing can help address some of the challenges of aging populations by offering services such as fall detection, remote health monitoring and cognitive support performing activities of daily living for older adults. These technologies often explicitly support human values such as safety, well-being and autonomy. At the same time they raise questions about their impact on values such as dignity and privacy. As a result, designers of pervasive healthcare technologies are faced with a range of recurrent value issues to take into account during design. Despite their recurrence, there is little guidance available on dealing with these issues in design. Where these issues are successfully addressed in design, much of the resulting design knowledge is often lost because it is not captured for reuse, leaving projects to "reinvent the wheel". Design patterns are one attempt to capture and document proven solutions. Until now value issues have not been systematically addressed in existing design patterns. In this paper, we propose a method to discover design patterns for explicitly supported values in pervasive health computing and give several examples of patterns.

3.1. Introduction

Pervasive computing has many promising applications for healthcare. Emerging pervasive computing technologies can help ameliorate many of the issues faced by societies with aging populations. The cognitive and physical decline associated with aging increases the need for informal or formal care. The strain this places on informal caregivers can drive the decision to place elderly family members in nursing homes. As populations age, the strain on healthcare systems, from nursing homes to physicians, grows as they struggle to meet the increased demand. This strain, along with rising health care costs, makes it desirable for elders to stay and age in their own homes as long as possible. Living independently is not without its risks though. Pervasive computing technologies can help address some of the challenges for seniors living independently by offering services such as fall detection, remote health monitoring and cognitive support performing activities of daily living.

The main goals of these technologies often explicitly support human values such as safety, well-being and autonomy. Fall detection systems aim to increase the safety of elders living alone by protecting them from harms associated with unattended falls. Activity recognition and prompting systems improve the independence of people with cognitive impairments (such as memory loss associated with mild dementia) by recognizing when a person is stuck performing a task and providing that person with the instructions or reminder needed to complete the task, thereby reducing the need for caregivers' assistance or even institutionalization. The same technologies also raise questions about their impact on values such as dignity and privacy. As a result, designers of pervasive healthcare technologies are faced with a range of value issues to take into account during design, in addition to technical and interaction challenges.

Many of these issues, whether technical, pertaining to interaction or values, are common to several pervasive computing technologies. That is, they are recurrent issues, and need to be addressed time and again. Despite their recurrence, there is little guidance available on dealing with these issues in design. Where these issues are successfully addressed in design, much of the resulting design knowledge is often lost because it is not captured for reuse, leaving projects to "reinvent the wheel".

Attempts have been made to help design processes address such challenges by discovering and documenting proven solutions to recurrent problems as design patterns. This work has focused on technical issues (e.g., context awareness [Riva et al. 2006]) and interaction design (e.g. [Landay et al.]). Value issues in pervasive computing, such as safety (of homeless youths) and surveillance values in participatory sensing have received attention in [1, 2]. Although there are pattern collections including patterns that deal with certain values such as privacy (e.g., [3]), value issues have not received the same attention technical and interaction issues have in design patterns for pervasive computing. Design patterns that focus on values seem a promising way of sharing and learning about best practices in supporting values and dealing with value issues in designing pervasive computing technologies for healthcare.

Discovering design patterns that address value issues is not trivial; it requires

some understanding of values and their relationship to technology. Though many pervasive healthcare technologies explicitly support certain values, this support is expressed in higher level project aims or goals. These values are rarely analyzed or defined in detail, nor are their relationships to the components that make up pervasive healthcare systems. To discover design patterns for values in pervasive health, we need to identify values in existing solutions, analyze their relationship to the components of those solutions, and identify patterns across design cases.

In this paper, we propose a method to discover design patterns for explicitly supported values in pervasive health computing. Our approach introduces a technique to identify values in existing pervasive healthcare systems documentation (here, research papers) by examining value statements. Identified value statements and architectural models of these systems are a starting point to analyze the relationships between system components and explicitly supported values. The relationships we identify form the basis of design patterns.

The rest of the paper is organized as follows. In Section 2, we discuss related work on values in pervasive computing, techniques to deal with values in design, and design patterns in pervasive computing. In Section 3, we present our approach. In Section 4, we apply our approach to a body of existing pervasive healthcare technologies and present the resulting patterns. In Section 5, we discuss the implications of this approach. In Section 6, we draw some conclusions, and offer suggestions for future work.

3.2. Related work

3.2.1. Values and Technology

The need to support human values through design of technology has initially emerged within four important research areas, i.e. Computer Ethics (e.g. [4]) and Social Informatics (e.g. [5]), which mainly analyze existing technologies and their intersections with human lives, and Computer Supported Cooperative Work (see e.g. [6]) and Participatory Design (e.g., [7]), which are more active approaches incorporating values such as collaboration and democracy into technology respectively. ([8] provide a more detailed description of these approaches).

In Human Computer Interaction several design methodologies have recently been developed to account for human values in the design process of new technology, for instance, worth-centered design [9], values in design [10] and value sensitive design [11]. Worth Centered Design focuses on worth for the user by "designing things that will motivate people to buy, learn, use or recommend an interactive product, and ideally most or all of these" ([9, p. 106]; see also: [12]). More concerned with moral and political values are the latter two methodologies. Both provide frameworks for bringing values into the process of designing technology. In the following we concentrate on the investigations that are part of the value sensitive design framework, and discuss recent approaches to identifying values.

Value sensitive design

Value Sensitive Design (VSD) aims to incorporate knowledge of the ethical impact of a technology into the design process. VSD "is a theoretically grounded approach to

the design of technology that accounts for human values in a principled and comprehensive manner throughout the design process" [8]. To that end, it provides an iterative three-partite methodology consisting of conceptual investigations, empirical investigations, and technical investigations, which can be conducted in any order and in an integrated fashion.

The conceptual investigation focuses on "values discovery and informed analyses of these values and potential value tensions" [13]. Several of the analyses of the conceptual investigation phase need to be informed by empirical investigations of the technology's context, and to evaluate particular designs. Technical investigations focus on the properties and mechanisms of existing technologies that either support or hinder certain human values. Alternatively, technical investigation can consist of designing a system to support identified human values. One example of this pro-active technical investigation is the design case described in [14]on usable security for implantable cardiac medical devices.

Value identification

Identifying, discovering or eliciting values from direct (i.e., users) and indirect stakeholders (i.e., people affected by a technology although they are not users) is a topic of ongoing research in VSD. VSD does not prescribe a specific method for empirically eliciting values from stakeholders, but states that "the entire range of quantitative and qualitative methods used in social science research is potentially applicable" [11]. While this range is broad, researchers have recently described the need to develop methods that are more specifically focused on identifying values in design [15]. Identifying these values entails, discovering the values at play in the current design context, contextualizing abstract values, understanding value dimensions, and identifying possible value tensions. One way to start investigating values in a design process is to look for value heuristics or value lists that, as [8] stated in an early VSD publication"have a distinctive claim on resources in the design process." In this early chapter the following (not comprehensive) list of values was identified to be of ethical import in technology design: Human Welfare, Ownership and Property, Privacy, Freedom From Bias, Universal Usability, Trust, Autonomy, Informed Consent, Accountability, Identity, Calmness, and Environmental Sustainability.. Surely, one can see how several of these relate to pervasive health applications (e.g., welfare, privacy or autonomy in ambient assisted living technologies).

Later Friedman and colleagues [11] revise their statement on this list and name the same list as "intended as a heuristic for suggesting values that should be considered in the investigation". While such lists can be utilized as a starting point for designing technology that supports human values, they have also been criticized by others. For instance Borning and Muller recently suggested, "that it would be better, when presenting such lists, to be explicit about the particular culture and viewpoint in which they were developed, rather than omitting this context and in the process making implicit claims about more universality than are warranted" [16]. Earlier, it was stated that there is a potential risk of privileging known values over value discovery (e.g., [15]), i.e. that early and rigidly articulated conceptual investigations may predispose researchers from hearing other value considerations arising during empirical investigations. Mitigating this risk could be done by a stronger focus on

understanding stakeholders' values in the context in which they are relevant and how they guide judgment in specific contexts. [15] propose a variation of Photo Elicitation Interviews as an empirical method to discover the relevant values in a design context. The use of photographs evokes feelings and memories, which can give clues to the importance of certain values situated in real life, thereby providing a contextual perspective on abstract values. Another way to discover values and their meanings in a certain context , which is less obtrusive and less prone to social desirability bias than interviews, is to conduct content analysis on texts, such as offline documents (e.g. research reports) or online texts (e.g. tweets as in [17]).

Value Categories

The role of values in design is studied in diverse research areas, and it is therefore not surprising that this role is discussed differently in these fields. However, this leads to complications when drawing conclusions from different work in these fields and applying them in the future. To put it simple, sometimes values are seen as "values are identifiable entities that appear in technologies, built in consciously or unconsciously by designers and concretized through affordances" [18] and sometimes they emerge from human actors. In social sciences values are the sources of people's behaviors and judgments. Here the questions arise, whether values are attributes of people, how they turn into features of technical systems or how they are mediated by technology. Practical guidelines to approach values in design would be useful at this point.

To that end [18] proposed a framework that supports researchers in getting a grip on values in the design and use of technology. They define two broad value categories to be distinguished : dimensions of values describing the source of values and dimensions describing the characteristics of the values themselves. To understand the source of values three dimensions can be distinguished: "state, which considers the construction of the source on a continuum from natural [e.g. people] to designed systems [e.g. technology], unit, which considers scale, ranging from individuals [people or technologies] to collective groups [societies or systems]; and finally assemblage, which describes the convergence of the group holding the values, from homogenous to hybrid groups of actors [whether people or technologies]." To understand the characteristics of values three dimensions can be distinguished: salience, (i.e. the import of values in a given setting ranging from peripheral to central), enactment (ranging from potential to performed) and intention (ranging from accidental to purposive) . How salient a value is depends on the source of the value and the context (privacy may be important for an individual in one setting but not in another). Some of these values may be potential (e.g. values designers would like to built into the systems, but which are not due to other constraints, while others are actually performed, i.e. values that a system materializes in the world. The latter can be accidental or purposive, i.e. deliberately built into a technology by its designers and materialized through affordances and policies.

This distinction is similar to Borning and Muller's distinction "among explicitly supported values, stakeholder values, and designer values. These are respectively values that are explicitly supported and adopted in a given investigation or project, values that are important to some (but not necessarily all) stakeholders, and the

designer or researcher's own personal values." In our work on design patterns we will focus on explicitly supported values because these are values that are identified and decided to be adopted in a project and subsequently embedded or accounted for in the technology design. Thus the translation from identified values and specific design qualities or functionality of a system has to be made. Unfortunately, VSD provides little general guidance, except providing a range of case studies, on how to do this. In pervasive health computing often this step is even left implicit. Design patterns are one way to bring this tacit knowledge to the surface and analyze it across many design cases to identify proven solutions.

3.2.2. Design patterns and pervasive computing

Design Patterns

Design patterns describe recurring (design) problems and the qualities that (proven) solutions to those problems have in common. The notion of design patterns originated in the field of architecture with the work of Christopher Alexander in the 1960s and 1970s [19] and made its way into software engineering, most popularly in the work of Gamma and colleagues [20]. From there, design patterns have made their way into the field Human-Computer Interaction (e.g., [21, 22]) and ubiquitous computing (e.g., [23]).

The aims of design patterns vary from field to field. Part of the purpose of Alexander's patterns was to encourage public participation in architectural design [19], and allow for diversity of solutions [24]. By contrast, the Gang of Four [20]book that introduced the popularized the design patterns software engineering specified its patterns in great detail, which suggests they did not aim for diversity and user involvement, but dissemination of knowledge among programmers [25].

Despite diverse aims, most approaches to design patterns share the characteristic of capturing a recurring problem and the essence (or invariant) of its solution. A design pattern's problem describes the major issue that the pattern addresses. Design patterns often also describe the context in which the problem occurs, which helps understand the problem and under which conditions the solution is applicable. The solution description expresses the qualities that proven solutions to the problem have in common. Solutions strike a balance between being general enough to be reused, and detailed enough to guide design. Patterns at different levels of abstraction can form a pattern language that designers can adapt to their project's particular level of complexity [26]. To illustrate the solution, design patterns present examples or known uses of the solution in practice.

Design Patterns in Pervasive Computing

In software engineering, the concept of design patterns was initially noticed in the context of research on code reuse. Such design patterns describe technical problems and solutions developers encounter in designing software. In a sense, the scope of these patterns is confined to software systems, and does not involve the world outside that software.

Some early work on patterns in software engineering did include solutions for user-interface design. These solutions fell within the scope of software systems

and were stated in terms of suggested code structures [19]. In contrast, interaction design patterns that emerged in Human-Computer Interaction often describe problems that concern the humans interacting with a system. They state problems in the domain of human interaction issues, and their solutions are stated "in terms of perceivable interaction behavior" [19].

In pervasive and ubiquitous computing, some work on design patterns focuses on technical problems and presents solutions in terms of code or hardware structures of components and systems [27, 28]. For example, Riva and colleagues' [27] design patterns to support context-awareness include the Enactor pattern. The intent of this pattern is to "provide application-driven support to context control and monitoring." They present the following motivation for the pattern:

"If control, monitoring, and action-triggering logic are merged into the same context management component, it is quite difficult to enable applications to inspect and control its execution state."

This problem focuses on technical challenges and does not directly concern users' interactions with a system. Part of the solution states:

"Enactors are designed to allow developers to easily encapsulate application logic in a component. [...] An Enactor has References that acquire data from Context sources, Listeners that monitor all changes, and Parameters that allow control."

The solution focuses on technical aspects clearly concerning developers and not users.

Other work emphasizes interaction with pervasive systems, identifying problems that concern users and solutions that involve human interactions with systems [25, 26]. For example, [26] includes a Context-Sensitive I/O pattern. It presents the following problem:

"Ubicomp devices will be used in a variety of locations and situations, but the device interfaces must not interrupt or distract the user from performing a primary task or annoy a nearby group of people."

This problem statement concerns user's interaction with ubiquitous computing technology. Moreover, it includes other people potentially affected by the system without using it. Again, the solution concerns human users:

"Input and output modalities should adapt to the user's current context."

Other work in this area falls somewhere in between, describing a problem that concerns users, but placing less emphasis on human interaction in the solution. For example, Roth's mobility patterns [29] include The remote proxy pattern. An excerpt of the context states: "consider a user browsing the web with a handheld device. The screen resolution of such a device is currently very poor, thus elements such as graphics and tables are difficult to display." The forces the pattern describes include: "a user who requests a specific task wants to save network bandwidth; wants to save computational resources (e.g. memory) on the local device, and; expects appropriate input/output behavior according to the locally available capabilities." The first part of the solution states: " the device does not connect directly to the network, but asks another device or computer to perform these tasks. This other computer, called the proxy accepts service requests from other devices;

connects to the actual service provider and performs the requested tasks; processes the results, and; sends them back to the initiating device." The pattern describes consequences of this solution, including: "Very often, devices used by the end-users have only poor capabilities. Nevertheless, users want to execute demanding tasks. As the greatest benefit, we do not have to change the public interface. The rest of the network remains unmodified, i.e. we can use the same infrastructure and protocols." The context and problem involve user's interaction with mobile devices, but the pattern's solution almost exclusively describes hardware and software components, and relationships between them. The consequences consider both the end-user and the structure of the system itself.

Design Patterns for Values

As we mentioned, in addition to technical and interaction design challenges, pervasive computing faces challenges pertaining to the impact pervasive computing systems have on human values. Some collections of design patterns for pervasive computing do include patterns that address issues with values such as privacy. For example Chung and colleagues' pre-patterns for ubiquitous computing dedicates a group of 15 patterns to "Developing Successful Privacy" [3]. However, values are not the primary focus of these collections, and many of the values pervasive computing systems explicitly support time and again, such as independence, social well-being and dignity, do not figure into patterns.

Within Value Sensitive Design, there has been some attention for design patterns [30]. However, these patterns describe solutions concerning the design process, rather than system components and their relation to stakeholders and their values. Dearden and Finlay suggest that design patterns reflect design values, in that selecting and recording patterns are value-laden activities that reflect the priorities and motivations of the patterns' writer [19]. They argue that these values are apparent in patterns in different ways, informing identification of 'good' design from which to discover patterns; influencing the selection of, and rationale provided for individual patterns; and the way in which they are used.

We agree that values can become inscribed in patterns in these ways, though we question how readily identifiable these values are in patterns, if they are not explicitly stated. Furthermore, Dearden and Finlay do not seem to acknowledge the role values play in the solutions themselves. As we argue above, technologies can support certain values while hindering others. Technologies provide certain "value suitabilities" [11] that design patterns capture by deriving solutions from practice.

3.3. Method

We propose using design patterns to capture design knowledge about value issues surrounding pervasive computing technologies for healthcare and their solutions for reuse. Like much of the work on design patterns described in the previous section, we examine existing examples of design to try to identify good practice and the distinguishing characteristics (or invariants) that make identified solutions successful. Here, we use academic literature on pervasive computing as a source

of examples of existing designs. These articles were collected in a literature survey described in [manuscript submitted for publication].

Our approach captures value issues explicitly and identifies their relationship to both problem and solution. Accordingly, the next step is to identify values from the collected examples of existing designs. As mentioned, there are different types of values in design: explicitly supported values, designer values, and stakeholder values. Here, we are limited to those values mentioned in the source articles, which we consider explicitly supported values.

Rather than base value identification on heuristic lists of values like those described in Section 2.1.2, we turn to the source articles themselves to identify value statements. These statements express a "pro-attitude" or favorable attitude (e.g., respect, admiration) towards an object. We opted to focus on value statements, because limiting ourselves to a heuristic list we would run the risk of overlooking values. On the other hand, using a broad definition of values as "what a person or group of people find important in life" risks identifying too many values.

To find value statements, we searched for sentences containing verbs that express pro-attitudes, such as aid, approve, benefit, foster, increase, improve, support, and others. These verbs often express or imply a favorable attitude toward some object (e.g., "I admire his honesty"). That is, such statements suggest that their object valued. We refer to these objects as value objects.

We listed all pro-attitude verbs and associated value objects we encountered in our selection of articles. As our focus here is on moral or personal values, we did not consider value objects that are qualities of the system, such as scalability, robustness and efficiency. Such qualities refer to properties of the system (e.g., the system's ability to change in size). This yielded a list of valued objects and associated articles.

We grouped the resulting value objects based on their similarity. For example, autonomy and independence both can refer to freedom from external control, so we grouped these as "independence". We used the resulting list of value categories to label each of the value statements (pro-attitude and value object) and corresponding source articles.

We examined the articles associated with each value group for mention of any values issues or challenges to the value in question. For example, an article containing the value statement "facilitate independent living" asserts that "falls are …a major obstacle to independent living" [31]. Based on this, we identified value-related problem statements for each system in each value group. This contextualize each system's explicitly supported values.

The next step was to identify the relationship between each system and its explicitly supported values. To this end, we took a "reverse-architecting" approach based on [27]. This method consists of three steps. The first, reverse engineering, involves creating an overall architectural model of the system under analysis, based on source code or documentation. Since source code was not available to us, we used each article's description of a system as documentation from which to produce an architectural model. Most of the articles included descriptions and diagrams of their system's architecture. The second step, sub-system identification,

uses architectural models of the system identify issue-specific sub-systems; that is, system components and relationships designed to address a specific issue. In our approach, the value-related problem statements identified in the previous step provide these specific issues.

We then used this information to discover design patterns. The level of abstraction of the identified value categories, the issue-specific (or value-specific) components that the systems in a value category have in common are likely to be too general to yield useful patterns. So, we used the problem statements to refine the value categories. The problem statements helped us distinguish multiple issues for each value. Each issue (or value-subcategory) was addressed by multiple systems, increasing the likelihood of discovering design decisions applied in several systems. Following [27], for each issue, we identified components of the associated systems that addressed that issue to form a candidate pattern. If components of subsequent systems were similar, we identified common elements, and modified the candidate pattern accordingly. If these components were too dissimilar, we created a new candidate pattern. As a result, there could be multiple patterns addressing the same issue. To write the patterns, we used the common elements of each system's value-related problem statement to describe the problem the pattern sought to address. We described the pattern's solution based on the common elements of the components addressing the problem. Finally, we included examples of known uses with references to the articles describing the systems associated with the pattern. We also included a context section in the patterns describing the relevant stakeholders and environment.

3.4. Case study — Patterns

In the following we present two design patterns that were derived using the approach above. Each pattern consists of a short statement about the context, followed by the value it concerns, a problem statement related to that value, and the solution. Last, a number of examples from the literature are provided for reference.

Cognitive Support for Independent Task Performance

Context Elderly people, wishing to remain in their own homes and live their lives as they choose to.

Value Independence – being able to perform one's activities of daily living autonomously

Problem Cognitive impairments associated with aging, such as dementia, lead to short term memory loss and difficulties performing activities of daily living such as preparing food, cleaning, and taking medication. As a result, people facing cognitive impairments must rely on others for assistance or even face institutionalization, hindering their independence.

Solution When someone with a cognitive impairment forgets to perform a (regular) task, forgets how to perform a task or gets stuck performing a task, provide him/her with the information s/he needs to successfully complete the task or activity. Use sensors to detect the person's proximity to or interactions with objects involved in performing the task. Use reasoning techniques to infer which task the person has forgotten to perform, or where in the activity s/he is stuck and remind or

prompt him/her. Or, provide instructions or reminders associated with the object.

Examples [32]employs descriptions of objects and actions involved in performing everyday tasks (Interaction Unit analysis) and a specification of available sensors deployed in areas and on objects involved in activities of daily living. The system uses a partially observable Markov decision process to model and recognize activities. When it detects that a user is stuck performing a (recognized) activity, it prompts the user through the task.

Non-Intrusive Activity Pattern Monitoring for Independent Living

Context Older adults living in their own homes face age-related chronic and degenerative diseases, risk of injury, and other risk factors that can lead to hospitalization or institutionalization. Timely detection of these diseases and risk factors can help lessen their harmful consequences. Information about a person's patterns of living and usual behaviors provides an indicator of his or her health status, and deviations from these patterns can indicate health decline.

Value Independence. Being able to live one's life as one chooses to, often in the comfort of one's own home.

Problem Capturing a person's behavior and patterns of living, and detecting deviations involves observing or monitoring him or her frequently enough to learn his or her routines and to catch deviations. Such monitoring would require that caregivers regularly visit the older adult, or that the older adult frequently visits caregivers, or is institutionalized. Technologies to support monitoring often use intrusive sensors to collect and present simple statistics. Factors including social or environmental factors, or biological rhythms affect a person's behavioral patterns, and make them difficult to assess and diagnose. As a result, a care provider often needs to be involved in the system loop. All of these issues diminish the older adult's ability to pursue her life as she chooses to within her own home, and thus hinder independence.

Solution A person's patterns of living consist of daily habits or repetitive activities of daily life. These activities can be characterized by repeating patterns of a person's movement through space over time. Non-intrusive motion sensors gather data about the older adult's movement through her house, registering times and locations at which motion was detected. These data provide a basic indication of room occupancy through the house. Automatically juxtaposing the relative mobility levels in each space (during a specific time interval) over the whole day reveals a 24 hour activity profile or pattern. Behavioral anomalies are detected from relative mobility levels in a specific space and time that are outliers compared to the total of relative mobility levels in that space. When a behavioral anomaly is detected, caregivers can be involved. This solution lets older adults remain in their own homes, only involving caregivers and others when needed.

Examples The BehaviorScope system uses non-intrusive passive infrared sensors to detect the user's movement through the house [33]. The system's intelligent gateway collects motion statistics and determines the relative mobility levels throughout the user's house. It detects repeating patterns of activity to extract a model of the user's daily habits and learn the temporal characteristics of those activities to improve accuracy. The system allows users to specify rules for be-

havior interpretation in terms of spatial and temporal characteristics of behaviors and sensors that will detect these spatial and temporal features. These rules help achieve activity recognition at different levels of spatial and temporal granularity. They allow the system to generate relevant activity summaries for caregivers. Users can also set alarm triggers, so that the system can notify caregivers in case of an emergency or behavioral anomaly. The system's non-intrusiveness and automated monitoring and activity recognition help elders maintain their independence by letting them remain in their own home and reducing their dependence on caregivers to the most essential interactions.

3.5. Discussion

Up to this point we have argued for the benefits of value sensitive design patterns for pervasive health computing, we have provided a method to derive them, and we have given examples of what these patterns would look like. These were the crucial steps in our endeavor to capture knowledge about the relationship between values and technology, and in particular to provide knowledge about proven solutions to recurrent value issues in pervasive health systems and how to explicitly support them in design. Several aspects of this work remain the topic of further research, (1) including the possible sources for value sensitive design patterns, (2) the identification of values and value tensions, as well as the (3) selection of cases/solutions to be included in the creation of design patterns.

Regarding the first point, the work presented here is limited to the investigation of a body of research papers that present solutions within pervasive health computing, in particular for ambient assisted living. Research reports, however, often do not tell the complete story of how a system was designed and we have to be aware that they are written by certain people, i.e. researchers, who may or may not be the developers of the concrete technologies, and with a certain goal, i.e. to publish their results in academic outlets. Often such reports fall short an in-depth description of the complete design and development process due to the restrictive page limits and therefore only provide a window to the whole process. To capture the complete process we need to expand the analysis of values and their consideration in the design from grant proposals, design briefings, work-in-progress reports, and final documentations. Furthermore, not only written texts should be analyzed but also statements and decisions made by the design team throughout the process, as well as the resulting artifacts.

Regarding the second point (identification of values), we developed and presented one way to discover value in texts without determining the values beforehand, but rather looked for value-laden statements that would later lead us to the values. Heuristic lists of values, such as those developed within the VSD framework [11] or Cheng and Fleischmann's Meta-Inventory of Human Values [34], can also be used as a starting point to identify values, providing definitions and references to relevant literature. Although as mentioned earlier, we have to be aware that this procedure may create a possible bias to values in these lists and ignore others, we still consider it as an alternative of high utility, especially in cases where designers and developers are less familiar with dimensions of human values and their rela-

tions to technological artifacts. In addition, values rarely occur in isolation but are part of complex value systems of individuals, groups of people, and socio-technical systems. It is unavoidable that tensions between values occur in these systems. In our work so far we have focused on single explicitly supported values in pervasive health system design. A next step should include the analysis of value tensions that commonly occur, in particular between explicitly supported values to be embedded in technical functions, and stakeholder values.

Regarding the third point (selection of cases/solutions), it is important to note, that not each solution that has been published should be included in the identification of design patterns. Design patterns should represent successful solutions to common problems. Therefore, it is necessary to judge the quality of solutions presented in the literature. One way is to look only at solutions that have been evaluated by end-users, and preferably in a real-life context. Ideally solutions should be implemented in consumer products or at least high-fidelity prototypes that are tested in long-term studies, to judge whether the solutions really bring the expected health benefits. This, however, is challenging, as many research projects are only evaluated in limited settings and often do not make it one-to-one into consumer products. Finding quality indicators of solutions despite these restrictions is a challenge to be tackled in future research.

Last, we would like to point out that value sensitive design patterns not only serves as concrete tools for developers in designing technical solutions for pervasive health, but also hold promise as a means to make the complex and commonly tacit knowledge about how values and technologies relate more transparent in the design of solutions and thereby educate people on values in design.

3.6. Conclusion

It has been established in several research domains that values are implicated in technology design and technology use. In pervasive health computing technologies often explicitly support human values such as safety, well-being and autonomy, but sometimes they also hinder other values such as privacy or responsibility. Despite their recurrence in many projects, little of the often tacit knowledge of how values are embedded in the design process and final artifact is captured in a way that it can be easily reused across design contexts. This poses challenges for designers of these technologies in dealing with values to be explicitly supported, values that stakeholders hold and value tensions that arise, and often they have to "reinvent the wheel". Design patterns are a way to capture this tacit knowledge, but attempts to discover and document proven solutions to recurrent problems as design patterns have so far mostly focused on technical and interaction issues, while value issues have not received this attention. We argued, that design patterns that focus on values are a promising way of sharing and learning about best practices in supporting values and dealing with value issues in designing pervasive computing technologies for healthcare. In this paper we proposed a method to derive such value sensitive design patterns from a large body of pervasive health literature and presented several examples of design patterns to be used. This work is not only seen as way to provide guidance in the design of a concrete system, but also as a

way to teach designers and software engineers how values and technology relate. We envision these to be expanded to a large set of value sensitive design patterns for pervasive health computing, also focusing on value tensions, and ultimately to be further developed into a pattern language in the future.

References

[1] J. Woelfer and D. Hendry, *Designing ubiquitous information systems for a community of homeless young people: precaution and a way forward,* Personal and Ubiquitous Computing **15**, 565 (2011).

[2] K. Shilton, *Values levers: Building ethics into design,* Science, Technology & Human Values **38**, 374 (2013), http://sth.sagepub.com/content/38/3/374.full.pdf+html .

[3] E. S. Chung, J. I. Hong, J. Lin, M. K. Prabaker, J. A. Landay, and A. L. Liu, *Development and evaluation of emerging design patterns for ubiquitous computing,* in *Proceedings of the 5th Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques*, DIS '04 (ACM, New York, NY, USA, 2004) pp. 233–242.

[4] T. W. Bynum (Ed.), *[entire issue],* Metaphilosophy **16** (1985).

[5] R. Kling and S. L. Star, *Human centered systems in the perspective of organizational and social informatics,* SIGCAS Comput. Soc. **28**, 22 (1998).

[6] L. Fuchs, *Area: A cross-application notification service for groupware,* in *ECSCW '99*, edited by S. Bødker, M. Kyng, and K. Schmidt (Springer Netherlands, 1999) pp. 61–80.

[7] P. Ehn, *Work-oriented design of computer artifacts*, Ph.D. thesis, Umeå University (1988).

[8] B. Friedman and P. H. Kahn, Jr., *Human values, ethics, and design,* in *The human-computer interaction handbook*, edited by J. A. Jacko and A. Sears (L. Erlbaum Associates Inc., Hillsdale, NJ, USA, 2003) Chap. 61, pp. 1177–1201.

[9] G. Cockton, *Designing worth is worth designing,* in *Proceedings of the 4th Nordic Conference on Human-computer Interaction: Changing Roles*, NordiCHI '06 (ACM, New York, NY, USA, 2006) pp. 165–174.

[10] M. Flanagan, D. Howe, and H. Nissenbaum, *Embodying values in technology: Theory and practice,* Information technology and moral philosophy , 322 (2008).

[11] B. Friedman, P. Kahn Jr, and A. Borning, *Value sensitive design and information systems,* Human-computer interaction in management information systems: Foundations **4** (2006).

[12] G. Cockton, *Feature: Designing worth—connecting preferred means to desired ends,* interactions **15**, 54 (2008).

[13] J. K. Miller, B. Friedman, and G. Jancke, *Value tensions in design: the value sensitive design, development, and appropriation of a corporation's groupware system,* in *Proceedings of the 2007 international ACM conference on Supporting group work*, GROUP '07 (ACM, New York, NY, USA, 2007) pp. 281–290.

[14] T. Denning, A. Borning, B. Friedman, B. T. Gill, T. Kohno, and W. H. Maisel, *Patients, pacemakers, and implantable defibrillators: Human values and security for wireless implantable medical devices,* in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10 (ACM, New York, NY, USA, 2010) pp. 917–926.

[15] C. A. Le Dantec, E. S. Poole, and S. P. Wyche, *Values as lived experience: evolving value sensitive design in support of value discovery,* in *Proceedings of the 27th international conference on Human factors in computing systems*, CHI '09 (ACM, New York, NY, USA, 2009) pp. 1141–1150.

[16] A. Borning and M. Muller, *Next steps for value sensitive design,* in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12 (ACM, New York, NY, USA, 2012) pp. 1125–1134.

[17] J. A. Koepfler and K. R. Fleischmann, *Studying the values of hard-to-reach populations: Content analysis of tweets by the 21st century homeless,* in *Proceedings of the 2012 iConference*, iConference '12 (ACM, New York, NY, USA, 2012) pp. 48–55.

[18] K. Shilton, J. A. Koepfler, and K. R. Fleischmann, *Charting sociotechnical dimensions of values for design research,* The Information Society (2013).

[19] A. Dearden and J. Finlay, *Pattern languages in hci: A critical review,* Human-Computer Interaction **21**, 49 (2006).

[20] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design patterns: Elements of reusable object-oriented design,* (1995).

[21] E. Bayle, R. Bellamy, G. Casaday, T. Erickson, S. Fincher, B. Grinter, B. Gross, D. Lehder, H. Marmolin, B. Moore, C. Potts, G. Skousen, and J. Thomas, *Putting it all together: towards a pattern language for interaction design: A chi 97 workshop,* SIGCHI Bull. **30**, 17 (1998).

[22] J. O. Borchers, *A pattern approach to interaction design,* AI & Society **15**, 359 (2001), 10.1007/BF01206115.

[23] T. S. Saponas, M. K. Prabaker, G. D. Abowd, and J. A. Landay, *The impact of pre-patterns on the design of digital home applications,* in *Proceedings of the 6th conference on Designing Interactive systems*, DIS '06 (ACM, New York, NY, USA, 2006) pp. 189–198.

3

[24] C. Alexander, S. Ishikawa, and M. Silverstein, *Pattern languages,* Center for Environmental Structure **2** (1977).

[25] A. Wilde, P. Bruegger, and B. Hirsbrunner, *An overview of human-computer interaction patterns in pervasive systems,* in *User Science and Engineering (i-USEr), 2010 International Conference on* (2010) pp. 145–150.

[26] J. Landay and G. Borriello, *Design patterns for ubiquitous computing,* Computer **36**, 93 (2003).

[27] O. Riva, C. Di Flora, S. Russo, and K. Raatikainen, *Unearthing design patterns to support context-awareness,* in *Pervasive Computing and Communications Workshops, 2006. PerCom Workshops 2006. Fourth Annual IEEE International Conference on* (2006) pp. 5 pp.–387.

[28] G. Rossi, S. Gordillo, and F. Lyardet, *Design patterns for context-aware adaptation,* in *Applications and the Internet Workshops, 2005. Saint Workshops 2005. The 2005 Symposium on* (2005) pp. 170–173.

[29] J. Roth, *Patterns of mobile interaction,* Personal and Ubiquitous Computing **6**, 282 (2002).

[30] B. Friedman, *Value sensitive design,* in *Liberating voices: A pattern language for communication revolution*, edited by D. Schuler (The MIT Press, 2008) pp. 366–368.

[31] A. Sixsmith and N. Johnson, *A smart sensor to detect the falls of the elderly,* Pervasive Computing, IEEE **3**, 42 (2004).

[32] J. Hoey, T. Plötz, D. Jackson, A. Monk, C. Pham, and P. Olivier, *Rapid specification and automated generation of prompting systems to assist people with dementia,* Pervasive and Mobile Computing **7**, 299 (2011).

[33] A. Bamis, D. Lymberopoulos, T. Teixeira, and A. Savvides, *The behaviorscope framework for enabling ambient assisted living,* Personal and Ubiquitous Computing **14**, 473 (2010).

[34] A.-S. Cheng and K. R. Fleischmann, *Developing a meta-inventory of human values,* Proceedings of the American Society for Information Science and Technology **47**, 1 (2010).

4

Ambient Intelligence Implies Responsibility

Ambient Intelligence Systems are complex, knowledge providing systems. The group of intended users typically comprises both various kinds of professionals and of laymen. In this chapter we address responsibility issues associated with the use and design of AmISs. We explain the connection between knowledge and responsibility, and use that connection to show that the usage of Ambient Intelligence Systems leads to moral responsibilities for all types of users (professionals and otherwise). It follows from the same reasoning scheme applied to the role of the AmIS developer, that she is responsible to design the system for responsible use. We give some initial criteria for such design for responsibility of AmISs.

4.1. Introduction

Ambient Intelligence Systems (AmISs) are being developed to improve the quality of life. However, as is typically the case with new technology, AmISs also introduce some new concerns. This chapter first argues that using an AmIS will burden every type of user with responsibilities and then studies the possible impact of these responsibilities on the types of users. This has an effect on the design responsibilities of the developer of AmISs. The argument is based on the idea that knowledge implies responsibilities; see e.g., [1**?**]. As human users of Knowledge Based Systems (KBSs) in general, and AmISs in particular, are equipped with more knowledge by using such systems, we address the question whether this leads to new or more responsibilities.

Ambient Intelligence Systems can be seen as a form of Knowledge-Based Systems (see e.g., [2]), where AmISs distinguish themselves e.g. by their distributed nature, and the way they are used (in time, place, and in numbers of users). AmISs have in common with KBSs that they provide the user with information and knowledge, though AmISs, where such systems are ubiquitous and invisible, can support users without users knowing. From the starting point that with knowledge comes responsibility, AmISs raise a number of questions: What kind of responsibilities come with the use of AmISs? For which users? Under which conditions? Could the impact be potentially more substantial than that of traditional Knowledge Based Systems? Can we single out aspects that are due to the inclusion of (Wireless) Sensor Networks and Intelligent Agents in AmISs?

AmISs typically are designed for a combination of both professional users and layperson users. For example, the CCF2 system to support independent living for the elderly (see e.g., [3]), has professional nursing staff, the elderly client, and family members of the client as users. These user groups together form the group of the intended users. To add to the variety within the user group, an AmIS may also have unintended users, i.e., people that the stakeholders and designers of the system did not envisage as users when they designed the system. Although issues of security (which are tied to unintended use) fall out of the scope of this chapter, we find it important to address the difference between intended and unintended users in terms of responsibilities.

The various user types, such as physicians, nurses and family members, and their differences in training and background puts the different types of users in different positions with respect to responsible use of the system. An important question is what these different responsibilities are for the different user types, and whether all users are aware of their responsibilities. In this chapter, we will argue that an important responsibility of the designer is to at least make all types of intended users of AmISs aware of the responsibilities that come with the use of the system, so that they can accept them (informed consent).

Another important question with respect to responsibility in AmISs is the role of intelligent software agents [4], though the question of the moral agenthood of intelligent software agents is beyond the scope of this work. Intelligent software agents typically play a role as representatives of human stakeholders, for example software agents that manage users' personal data or an agent that plans appointments with

a physician. Should we regard these agents as part of the design of the AmIS, or as users of it (as the human parties they represent)? If seen as users, they would inherit the moral responsibilities that come with the use of the system. If seen as part of the AmIS, responsibility would be attributed to their human designers.

The previous remark points at an important discussion this chapter intends to initiate, concerning the responsibilities of the AmIS developers. If responsibilities can be associated with the use of AmISs, then we claim this should in itself impact the design of AmISs. The question is, how? What are the criteria for designing a system for responsible usage?

The focus of this chapter is on how ambient intelligence systems affect their stakeholders' and designers' responsibilities through the knowledge these systems produce. Before we can examine these relationships, we must explain the relationship between knowledge and responsibility. This requires an explanation of what responsibility is; how it affects people and how it is ascribed to them. Therefore, this chapter is organized as follows. Section 1 discusses the notion of (moral) responsibility. Various types of responsibilities are discussed as well as when people can be held responsible (the so-called knowledge condition of responsibility). Section 2 relates knowledge-based systems to responsibilities. A case study is discussed in Section 3 with an emphasis on the stakeholders and their responsibilities. Section 4 discusses the knowledge produced by AmIS and the responsibilities that brings to the various user types. Section 5 reflects on the previous sections of the chapter and distills design responsibilities for the designers of Ambient Intelligence Systems. Throughout the chapter, the CCF2 system for Independent Living is used as an example AmIS.

4.2. Responsibility

Responsibility affects people in different ways. It can dictate how someone should behave; for example, if a nurse is responsible for the stability of a patient's condition, she should check on that patient frequently. Responsibility also co-determines how others should behave in light of someone's actions, e.g. whether they should (also) check the patient, or whether they should check the nurse. There are many different ways in which the concept of responsibility is used, and it is hard to define clearly under which circumstances and how people "get" responsibilities, i.e. how responsibility is ascribed. This section provides some forms of responsibility and a definition of fair ascription of responsibility.

4.2.1. Forms of Responsibility

There is no unified understanding of responsibility. In fact, there are many subtly different senses in which the word "responsibility" is used, as this anecdote illustrates [5, p. 96] (the anecdote originates from [6]): (1) Smith had always been an exceedingly responsible person, (2) and as captain of the ship he was responsible for the safety of his passengers and crew. But on his last voyage he drank himself into a stupor, (3) and he was responsible for the loss of his ship and many lives. (4) Smith's defense attorney argued that the alcohol and his transient depression were

responsible for his misconduct, (5) but the prosecution's medical experts confirmed that he was fully responsible when he started drinking since he was not suffering from depression at that time. (6) Smith should take responsibility for his victims' families' losses, but his employer will probably be held responsible for them, as Smith is insolvent and uninsured.

The six different forms of responsibility used in this example, are what Vincent calls (1) virtue, (2) role, (3) outcome, (4) causal, (5) capacity, and (6) liability responsibility. Virtue responsibility (1) refers to having the characteristic of being the type of person who does the right thing. Role responsibility (2) refers to what someone ought to do, i.e., the duties that come with a role; the captain ought to have seen to the safety of his passengers and crew. Outcome responsibility (3) refers to rightfully being identifiable as the cause of some situation X and deserving of praise or blame for it; in the example the captain is responsible for the loss of the ship and of many lives. This concept is closely related to causal responsibility (4) that refers to a causal chain that leads to the outcome that is situation X; in this case the alcohol that the captain should not have taken. Capacity responsibility (5) refers to the fact the captain had the capacity to realize that he shouldn't drink so much alcohol. Being capacity responsible means having the capability to understand what behavior relevant norms require, to reason about what one ought to do in light of such requirements, and to control one's behavior accordingly [7]. Liability responsibility refers to the responsibility of the captain for the consequences of the outcome, in particular for the financial losses of the families.

Accountability is closely related to the notion of liability. Whereas liability involves the question of who will pay for damages done or who will set things straight, accountability involves answering for one's actions (of which damages were an outcome) [8]. This concept is also related to Vincent's concept of causal responsibility, in that in answering for one's actions, there is assumingly a causal connection between those actions and the outcome. Also, there is a connection between the concept of outcome responsibility and accountability, in that "[a]n important set of cases in which one may reasonably expect accountability for a harm is that in which an analysis points to an individual (or group of individuals) who are morally blameworthy for it" [8, p. 28].

For other views of responsibility, such as those based on merits, rights or consequences, see [9]. However, this chapter focuses on those notions of responsibility that are most relevant for AmISs during design time and when starting to use the functionalities of AmISs.

For analysis of responsibility also the notions of "forward-looking" and "backward-looking" responsibility are relevant, see, e.g., [10]. Forward-looking responsibility refers to being expected or required to behave or act in a certain way. In other words if a person has a forward-looking responsibility, she has a duty to do X, she is bound to do X, and she ought to do X.

There is a relationship between forward-looking and backward-looking forms of responsibility, in that, under certain circumstances forward-looking responsibility implies backward-looking responsibility [10] or forward-looking responsibilities can be translated into backward-looking responsibilities and vice versa [9]. That is, if

one has a duty to ensure some state of affairs, X, and one does in fact ensure that X is the state of affairs, one would be morally responsible for X and praised for it. If one failed to ensure this state of affairs (and the situation was within one's control), one would be morally responsible and blamed for X not being the state of affairs.

Liability responsibility, i.e., being identified as the person (or legal entity) that will pay for the harm caused, involves both backward-looking and forward-looking elements. It is backward-looking in the sense that it is attributed on the basis of past causal contributions to outcomes (e.g., the sinking of the ship) but implies a duty to rectify matters (e.g., paying for the harm caused) [11].

In this chapter all notions of responsibility are relevant that can be ascribed to users of AmISs and the designers of AmISs, whether forward- or backward-looking. The next section discusses when responsibility can be ascribed.

4.2.2. Ascribing Responsibility

As this chapter focuses on the responsibilities that users and designers of AmISs can have, it is important to realize under which circumstances responsibility can be ascribed, which differs for different forms of responsibility.

Ascribing backward-looking responsibility (e.g., outcome or liability responsibility in Vincent's taxonomy), involves praising or blaming someone for a rightful or wrongful outcome to which her actions contributed. As such praise or blame can have consequences for the praised or blamed party, the ascription of this form of responsibility should be fair, cf. e.g. [9]. Doorn lists five conditions to be fulfilled in order for ascription of backward-looking responsibility to be fair (on a merit-based view of responsibility ascription), namely moral agency, voluntariness or freedom, knowledge of the consequences, causality, and transgression of a norm [9].

The moral agency condition is understood as the condition that the responsible actor possesses adequate mental faculties at the moment of engaging in an action, and thus is related to capacity responsibility. The voluntariness condition is the condition that the actor voluntarily performed the action that led to the outcome for which she is responsible. Taking this condition into consideration, actions performed under compulsion or other circumstances outside of the actors control are reasons to refrain from ascribing backward-looking responsibility to that actor.

The knowledge condition of responsibility as understood by Doorn is the condition that the actor knew or could have known the outcome of her actions [9], again related to capacity responsibility. We will discuss the knowledge condition of responsibility in more detail in the next section. The causality condition means that the actor's action or inaction is the cause or contributed causally to the outcome, i.e., causal responsibility.

Finally, the condition that there is a transgression of a norm entails that responsibility can only be ascribed fairly if the actor's action that causally contributed to the harmful outcome infringed on a relevant norm. The source of norms to be considered comes from either the role someone plays, or from the cultural setting, or from the law. Relevancy of norms is situational and is hard to describe in general terms. The ascription of forward-looking responsibility is somewhat different. As forward-looking responsibility is related to role and liability responsibility, identify-

ing conditions for fair ascription of forward-looking responsibility is a question of identifying sources of duties and obligations. [12] mentions four criteria for fair attribution of forward-looking responsibility: the agent does what she ought to do; the agent is able to do what she ought to do; the agent knows what she ought to do; the norm-violation is not unintentional.

Most of Mastop's conditions are intuitively clear; it seems unfair to obligate someone to do something if she is unable to do that thing. Similarly, it seems unfair to obligate someone to do something if she does not know she ought to do that thing. The not unintentional norm-violation condition seems to be the forward-looking dual of the voluntariness condition of [9].

However, the first criterion for ascribing forward-looking responsibility fairly, i.e. the agent does what she ought to do, requires some discussion. This criterion seems to suggest that forward-looking responsibility be ascribed at the moment the agent is performing the action that the responsibility requires. We argue that forward-looking responsibility refers to the obligation of behaving and acting in a responsible way. Note that the definitions of ascribing both backward-looking and forward-looking responsibility contain a knowledge condition. This is the fundamental reference as to why knowledge implies responsibility. As the focus of this chapter is on the relationship between knowledge and responsibilities, the knowledge condition and its limits are discussed in more detail in the next section. In this discussion, the other conditions are assumed to have been fulfilled.

4.2.3. The Knowledge Condition of Responsibility

Both ascriptions of backward-looking and forward-looking responsibility involve a knowledge condition. The previous section described under which conditions responsibility can be ascribed. Ascription can be done by others, but of course also by the person herself. When considering users and designers of AmISs the question is which are the responsibilities that can be ascribed to them? We use the following argument: If people know that certain responsibilities result from having a piece of knowledge, then if people have that piece of knowledge, the resulting responsibilities are also theirs, and they should be aware of the fact that they have those responsibilities. So, assuming all other ascription conditions have been met, we can schematize the argument referring informally to a logic including knowledge and responsibilities.[2] The first step is to conclude that a person having a piece of knowledge, in fact should carry the corresponding responsibility:

$\forall i : K_i(k) \rightarrow R_i(r)$ Knowing k implies carrying responsibility r

$K_a(k)$ a knows k

———————————

$R_a(r)$ a carries responsibility r

In order to be held responsible, as we argued, the person should also be aware of her responsibility. Above, we assumed that the people know that certain responsi-

---

[2]This is meant as a way of schematically presenting our argument, rather than as a fully formal account. We do assume the knowledge modality to be closed under elementary logical rules like modus ponens and universal instantiation, as is the case in standard epistemic logic. The R-operator serves as a predicate rather than a modality.

bilities result from having a certain piece of knowledge, this adds an extra layer in the assumptions:

$K_a(\forall i K_i(k) \to R_i(r))$ *a* knows that knowing *k* implies responsibility *r*
$K_a(k)$ *a* knows *k*
_____

$K_a(R_a(r))$ *a* is aware that she carries responsibility *r*

The question now is under which circumstances the preconditions of the above arguments can be taken to hold. Knowledge is a condition for responsibility in that responsibility can only be ascribed (assuming other conditions have been fulfilled) if the actor to which we seek to ascribe responsibility knew that his act would lead to a certain outcome and knew that this outcome was right or wrong. The first part of George Sher's Full Epistemic Condition (FEC) of responsibility is equivalent to this view. The Full Epistemic Condition of responsibility states that [13, p. 143]:

> When someone performs an act in a way that satisfies the voluntariness condition, and when he also satisfies any other conditions for responsibility that are independent of the epistemic condition, he is responsible for his act's morally or prudentially relevant feature if, but only if, he either 1. is consciously aware that the act has that feature (i.e., is wrong or foolish or right or prudent) when he performs it; or else 2. is unaware that the act is wrong or foolish despite having evidence for its wrongness or foolishness his failure to recognize which a) falls below some applicable standard, and b) is caused by the interaction of some combination of his constitutive attitudes, dispositions, and traits; or else 3. is unaware that the act is right or prudent despite having made enough cognitive contact with the evidence for its rightness or prudence to enable him to perform the act on that basis.

For example, a nurse in a care home, Donna, empties a bottle of rat poison into the ice machine in the recreation room. Donna knows that the air conditioning in the care home is faulty, so residents are likely to put ice in their beverages. Furthermore, Donna has read and understood the label of the bottle of rat poison, which clearly states that under no circumstances should human beings ingest the contents of the bottle. Over time, residents start to become ill (not dying, as the rat poison was diluted in the ice machine's water supply). Eventually, the care home physician finds out that this occurred as a result of poisoning. Most people would be inclined to blame Donna for the residents' illness. Assuming other conditions for ascribing responsibility have been met, Donna would be blamed because she knew that adding rat poison to something people would ingest would lead to their becoming ill. Furthermore, she knew that the outcome of such an act (and the act itself) was wrong.

Donna's actions exemplify the knowledge condition at work. Knowing that one's actions could lead to a certain outcome and knowing that that outcome is morally wrong are grounds for ascribing responsibility. It could be the case that one does not possess such knowledge in performing a certain act. In cases where one lacks the knowledge that one's actions will lead to a certain outcome and/or lacks the

knowledge that this outcome is wrong, we would intuitively excuse this person from responsibility on the grounds of ignorance. We take ignorance to mean "not knowing". For example, if another nurse, Horatio, brought a resident a pitcher of water filled with ice cubes, and later that day the resident died (after drinking the entire pitcher), we would not blame Horatio for the resident's death, as Horatio was ignorant of the fact that the ice cubes were poisoned.

There are, however, cases in which responsibility can fairly be ascribed to an actor, despite that actor's ignorance of the wrongness of her actions or the outcomes thereof. According to Sher's FEC, the actor who is unaware of the wrongness of her act is still responsible if she is unaware despite having evidence for its wrongness and this unawareness falls below some applicable standard and is caused by some interaction of her constitutive attitudes, dispositions and traits [13]. On a slightly different view, one can do some wrong act A, ignorant of the fact that it is wrong, and still be responsible for that act and/or its outcomes if at some earlier point one knowingly performed another wrongful act or omission of which the ignorance of the wrongness of A is a consequence. In Zimmerman's words, "culpability for ignorant behavior must be rooted in culpability that involves no ignorance" [1, p. 417]. In other words, the person to whom we attribute responsibility was at some point aware of wrongdoing. One source for such ignorance is negligence. Zimmerman defines a situation of negligence as a situation in which a person believes she is able to perform a certain set of precautionary measures against a negative outcome of an event; believes that not performing these events yields a possibility of a negative event occurring; unjustifiably decides not to do any of a subset of the set of precautionary measures; does an act A without doing or attempting to do any of a subset of the precautionary measures [14].

Furthermore, ascription of moral responsibility for negligence requires not just that one actually foresee the action and its consequences (currently believe that there is a possibility the event in question will occur at some later time), but that these are foreseeable. Zimmerman defines this foreseeability as follows. "[An event] e's occurring at [time] t2 is foreseeable (to some degree) by P if and only if it is possible that P foresees at t1 (to some degree) that e will occur at t2" [14, p. 207].

For example, a third nurse, Patsy, is bringing the residents their medication, and pouring each of them a glass of water with which to take their medication. Patsy stays with each patient to ensure that he finishes his medication. Upon drinking his glass of water, the first patient complains that he feels unwell. Patsy dismisses his complaint and proceeds to the second patient. The second patient has the same complaint, which Patsy also dismisses. Later that day, when all the patients Patsy saw are ill and it is discovered that they were poisoned, Patsy is blamed for ignoring patients' specific complaints and not recognizing the emerging pattern, both of which were signs that Patsy was giving them poisoned water.

One might argue that ascribing responsibility to an actor despite the fact that that the actor was ignorant of or did not advert to the risk created by her act or omission depends on claiming that that actor could have known (or could have done something). On this view, the claim that actor could have done something, but

actually did not, is a form of counterfactual reasoning and as such is problematic.

Such capacity judgments ("she could have known" or "she had the capacity to know") are indeed counterfactual, in that they 'suppose a world different than the actual world and ask, "what would have happened in that possible world?"' [15, p. 165]. However, we agree with Moore and Hurd that such counterfactual reasoning is inevitable in capacity judgments in that there are no 'unchanged-actual-world' capacity judgments with the exception of the judgment that "we all had the capacity to do just what we did and nothing else" (Ibid.).

This issue can be circumvented, to some extent, by focusing on the morally relevant factors that the type of capacity judgments of interest here depend on in explaining why an actor did not advert to a risk, such as "she would have adverted to the risk of her omission if she were not so selfishly indifferent to the welfare of other human beings and not so hurried to finish her rounds that she did not attend to signs that something was wrong"[3] . If the actor did not advert to a risk because, in this example, she was selfish, and if she can be blamed for being selfish, then we can construct counterfactuals in which the actor was not selfish and in those possible worlds would have adverted to the relevant risks. We can then infer that she could have adverted to the risk in the relevant sense [15]. The actor is then responsible based on the blameworthiness of her trait (selfishness) rather than a general capacity or incapacity to know her behavior was risky.

This, in turn, raises the question of whether an actor can be blamed for being stupid or having other flaws. Moore and Hurd discuss the blameworthiness of being clumsy, stupid, weak, and indifferent and draw a number of conclusions that are relevant in this context [15]. We argue with Moore and Hurd that while the actor's flaws might not be blameworthy in themselves, the actor's failure to take precautions against the harms that her flaws are predicted to cause or to change those flaws in ways that reduce risk to others is blameworthy. In such cases, the actor has failed to care sufficiently about the welfare of others in light of traits that have a known propensity to cause harm.

For forward-looking responsibility, matters are slightly different. Above, we argued that a criterion for fair attribution of forward-looking responsibility is that the responsible actor knows what she ought to do. So, the relationship between knowledge and forward-looking responsibility is that if an obligation is unknown to the actor to whom we wish to apply it, then that actor has no obligation (or forward-looking responsibility). Consider that the head nurse in the care home, Ellen, expects a new nurse, Django, to change all resident's bed sheets every morning at 11:00. However, Ellen has not told Django that he has this obligation. In fact, no one has told Django that he has this obligation. In this case, Django does not have the obligation to change all residents' bed sheets every morning at 11:00.

---

[3]This example is based on an example provided in [15].

4.3. Knowledge Based Systems Imply Responsibilities

The principle underlying Section 1.3, viz. that knowledge implies responsibility, has far reaching consequences for the use of information and communication technology in general, and more specifically for knowledge-based systems (KBSs). With the introduction and use of knowledge-based systems, the user has an additional source of knowledge and information. Thus, the use of a knowledge-based system brings more responsibilities to the user. In particular the user might have more obligations, and might also become liable for some possible situations. The obligations refer to actions that people should take (are expected to take) and liability is in light of actions they should have taken or should not have taken. Here actions refer not only to actions that affect the physical state of the world, but also to speech acts [16–18]. Speech acts such as informing and requesting can set things in motion in a way that is deemed responsible by the society.

Traditional KBSs, like medical expert systems, are typically designed to be used by (single) educated professional users in a clearly specified context of application. In contrast, Ambient Intelligence Systems (AmISs) are designed for more complex situations. AmISs have a number of features, such as sensitivity, responsiveness, adaptivity, transparency, ubiquity and intelligence [19]. These features have an impact on the relationship between AmISs and knowledge. Sensitivity, responsiveness, adaptivity and intelligence imply that AmISs reason about and act upon information they gather from their users and environment. Ubiquity implies that AmISs are applied in a diverse and possibly dynamic range of contexts. Transparency implies that (intended but also unintended) users will not always be aware of the systems they are subject to.

These features are knowledge-related and show that the responsibility implications of AmISs are more intricate than for traditional KBSs. First, there are more and more diverse users involved. For example, we can distinguish between the experts involved in the use of the system (professional users, e.g. doctors), and the lay-people (non-professional users, e.g. patients that are being assisted and monitored). Second, the ambient (or ubiquitous) feature of the technology, implies more flexibility in the contexts of application. Thus, chances are higher that there will be unintended users and usages of the system. It is up to the designer to see to it that the technology is good for the intended users and usages, and anticipate as much as possible the responsibilities that might ensue from using the system.

This means, that different responsibilities follow for the different types of users of an AmIS. Each type of users, has a different knowledge base. To simplify we distinguish simply between a professional user (e.g. the medical specialist) and a layman user (e.g. the patient). How the combination of this personal knowledge base with the knowledge base provided by the AmIS will also be different for different users, as their ways of combining the knowledge elements to a new combined knowledge base will be different: a professional can be expected to have a more sophisticated set of reasoning rules. Hence, there is a dependency of the combined knowledge base on both AmIS and user (U) as follows, as witnessed in the

subscripts of the following scheme[4]: $\overline{k_{AmIS}}, \overline{k_U} \vdash_{AmIS,U} \overline{k_{U,AmIS}}$

By combining the knowledge of the AmIS with the personal knowledge, information might be deduced, that is not deducible from either the knowledge base $\overline{k_{AmIS}}$ alone or from $\overline{k_U}$ alone. Thus, new responsibilities can arise that are neither the responsibility of the AmIS (alone), nor that of the user alone (i.e., when not using AmIS). As the human user is the only one (of *AmIS* and *U*) who is capable of inferring the new piece of combined knowledge, the user is the one that gains the responsibilities that come with inferred knowledge. So the use of AmISs leads to new responsibilities for its user.

It is important as a developer to be aware of this dependency when designing in the reasoning capacities of the AmIS: it becomes the responsibility of the developer to design the reasoning capacities fit for the right knowledge, reasoning capacities and intended use for each user type. Furthermore, the combined knowledge of the user, the knowledge of the developer of the knowledge base and reasoning capacity of the AmIS, give a combined responsibility for the responsible use of the AmIS. The designer is responsible for informing all types of users about the way the AmIS is designed, and what responsibilities follow from that knowledge for each (intended) user type.

Sher's Full Epistemic Condition, as discussed in Section 1, has an important implication for the designers of AmISs. With this chapter, we aim to make the designers aware of the responsibilities that the users of their AmISs will carry, and what this implies for the responsibility they themselves carry. In particular: by reading this chapter, designers will become consciously aware of the fact that the users of their AmISs will carry responsibilities, and thereby satisfy FEC. Remember the second argumentation scheme we presented in Section 1:

$$K_d(\forall i K_i(k_d) \rightarrow R_i(r_d))$$
$$K_d(k_d)$$
$$\overline{\qquad\qquad\qquad}$$
$$K_d(R_d(r_d))$$

Taking the designer as *d*, the contents of this chapter as $k \in k_d$, and the responsibilites of the designer for making users aware of their respective responsibilities as $r \in r_d$, we may conclude that we may ascribe the responsibility *r* to the designer of AmISs. For those designers who do not read this chapter, one could argue that prudence associated with their profession, requires them to read chapters like ours. This implies that designers reading this chapter can reasonably foresee that the use of the knowledge system they construct leads to responsibilities for themselves, and for the users of their knowledge system. Thereby they have the responsibility to design the AmISs in such way that (at least the intended) users will be made aware of the responsibilities that come with the use of the system (and that the user herself might not anticipate).

---

[4]We use the overline to indicate sets rather than single pieces of knowledge.

4.4. Independent Living for the Elderly: A Case Study

A prominent application domain for Ambient Intelligence is health care. The leading case study presented in the remainder of this chapter is based on the CCF2 project [3] that one of the authors was involved in, in which an AmIS was developed to support independent living for the elderly, and in particular to stimulate social and physical activity of elderly residents of a care home. This section describes the stakeholders in the CCF2 project, their knowledge and responsibilities prior to the introduction of the AmIS. The group of professional caregivers consists of general practitioners, nurses, occupational therapists, supervisors, and assistants to occupational therapists. Each professional is expected to know enough about the residents under their care to be able to carry out their profession. Of course they can be expected to have the types of knowledge required to practice their profession. Furthermore, they have their professional responsibilities for every resident (even those not directly under their supervision). Finally, as members of a social community and being of general good health and mental capacity they have general responsibilities for everyone in their surroundings.

From most general to most specific these responsibilities can be formalized in schemes. Starting with the general responsibilities of members of a social community, a scheme is set up that reflects general norms and their associated responsibilities.

Note that this is an operationalization of the definitions in Section 1. The rules can be applied in a forward-looking way to determine someone's responsibilities for now and the future. Also rule 2 can be used in backward-looking way to determine whether or not a person should have taken responsible actions at a time in the past. That means that current_state(S) should refer to what was current at that time in the past. For every person A that is a member of a social community C, if V is a value in the community, A knows (or should know) that V is value in C, and R is a responsibility associated with that value, and A knows (or should know) that R is a responsibility for value V, then A has that responsibility R.

  Member (A,C)
  Value_in(V,C)
  K_A(value(V,C))
  Responsibility_for_Value(R,V)
  K_A(Responsibility_for_Value(R,V))
  ——————————————————————- (Rule 1)
  Has_responsibility(A, R)

The following applies for any responsibility of any person: In a current situation S, if a person A knows (or should know that) some responsibility R applies in situation S, and P knows (or should know) that action α might be an appropriate action in light of responsibility R, then A is obliged to perform action α.

  Current_state(S)
  Has_responsibility(A,R)
  K_A(applies(R,S))
  K_A(appropriate(α,R))
  ——————————————- (Rule 2)

O_P(α)

To make it concrete, suppose a person p is a member of the social community of the care home, and consider the general value of fairness, for which a corresponding responsibility is to treat people fairly. Then person p has the responsibility to treat people fairly. Thus, if a situation arises where some resident is excluded from a game (without sufficient cause), and if p knows that speaking up for the excluded resident is appropriate, then p should speak up for the excluded resident.

The general responsibilities of professionals are more specific than the above. Every agent A having profession P has the role responsibilities R that come with her profession:

Is_a_profession(P)
Has_profession(A,P)
Role_responsibility_for(R, P)
———————————————— (Rule 3)
Has_responsibility(A,R)

For example, a nurse has the responsibility to provide medical assistance to a person in need for as far as her capabilities go, and call for help if needed. This can be put into effect by applying Rules 2 and 3 in combination with the following facts:

Is_a_profession(nurse);
Role_responsibility_for(medically_assist_those_in_need, nurse);
Role_responsibility_for(call_medical_assistance_if_needed, nurse);

For example, if a nurse sees that a resident has trouble drinking, and no other help is provided (which corresponds to the knowledge condition that she knows that her helping the resident to drink is an appropriate action), then she knows she should help the resident to drink.

Furthermore, a professional might have more specific responsibilities for those patients assigned to her care. For example, among the role responsibilities for occupational therapists is the duty to study the patient record of each resident H assigned to them, to set up a partial care plan for that resident H with respect to the professional's expertise and taking into account the partial care plans of other professionals for that resident, and to explain that partial care plan to the assistant S assigned to H so that assistant S can carry it out. Some of this is easily formalized according to the earlier schemes:

Role_responsibility_for(study_patient_record, occupational_therapist);
Role_responsibility_for(set_up_partial_care_plan, occupational_therapist);
Role_responsibility_for(explain_partial_care_plan, occupational_therapist;

However, per role responsibility more knowledge is required to adequately perform that role. For example, the role responsibilities to study a patient record, set up a partial care plan, and to explain a partial care plan should be performed only for the patients H assigned to the occupational therapist A.

Is_assigned_to(H, A)
Has_responsibility(A, R)
————————————————
Has_responsibility(A, R(H))

More specific knowledge is required to deal with studying a patient record (e.g., only study what you need to know to perform your role), or explaining a partial care plan. As an illustration:

Is_assistant_of_for(S,T,H)
Has_responsibility(T, explain_partial_care_plan(H))
_____
Has_responsibility(T, explain_partial_care_plan_of_to(H,S))

The main tasks of an assistant of an occupational therapist (o_t_assistant), as can be seen in the scenarios is receiving notifications of residents' level of physical activity and acting on this if necessary. For these tasks, the o_t_assistant has to monitor the physical activities, execute the partial care plan, and notifying the occupational therapist if the resident is not active enough for some time. Furthermore, the o_t_assistant is to stimulate active behavior by visiting the resident at least once a week and motivate the resident to schedule and participate in some activities.

Role_responsibility_for(monitor_physical_activities, o_t_assistant);
Role_responsibility_for(execute_care_plan, o_t_assistant);
Role_responsibility_for(notify_o_t, o_t_assistant);
Role_responsibility_for(stimulate_physical_activities, o_t_assistant);

The group of activity organizers is another group closely related to the group of professional caregivers in the CCF2 case. Their professional knowledge and skills include and pertain to organizing activities for individual needs and for groups. Their responsibilities include maintaining and making available a list of upcoming activities, organizing those events, and keeping track of which residents participate.

Finally, of non-professional caregivers only family members of residents were considered in the case study. Family members cannot be expected to have professional knowledge of care giving, although it cannot be excluded that they have such knowledge. Family members are attributed the general responsibilities that are attributed to any person of general good health and mental capacity. Furthermore, family members are expected to contribute and use their knowledge about the resident in terms of general health issues, preferences and values. The main duty of this group, in the case of elderly care, is to tend to the perceived needs of an old parent [20]. This can include assisting with activities of daily life (ADLs), such as bathing, dressing, moving around indoors, transferring from bed to chair, using the toilet or eating, and/or instrumental activities of daily life (IADLs) such as light housekeeping, meal preparation, grocery shopping, laundry, taking medications, managing money, telephoning, outdoor mobility and transportation. This is formalized directly in terms of having responsibilities, for example person A is a family member of resident H that has accepted the responsibility for bathing H:

Family_member_of(A, H)
Has_responsibility(A, bathing(H))

In the CCF2 case, (non-adult) grandchildren are specifically included in the family group of stakeholders. While not considered caregivers they interact with the system nonetheless, gaining and giving insight into their grandparents' preferences and daily activities.

4.5. Knowledge and Responsibility in Ambient Intelligence Systems

Ambient Intelligence Systems make use of and produce knowledge in to serve a variety of users. Their key features are sensitivity, responsiveness, adaptivity, transparency, ubiquity and intelligence [19]. The intelligence is typically spread over a number of contributing agents that by working together form an AmIS.

AmISs are sensitive in that they sense the environment in which they are situated. That is, AmISs obtain information about their operational environment. AmISs are responsive in the sense that they respond to (the presence of their) users, which is achieved, in part by modeling users, recognizing patterns, making decisions, and reasoning [19]. Adaptivity similarly involves reacting to information, in this case by changing (e.g., user models in the system). Transparency refers to the notion of the disappearing computer, that is, the idea that computing technology should weave itself into and thus become part of the environment [21]. The ubiquity of an AmIS means that it is present everywhere in the area where the AmIS is active. This ubiquity and the required intelligence make agent technology a driver for AmISs.

In all cases the AmIS is created for the use of some people, i.e., the direct stakeholders and might (inadvertently or intentionally) affect others, i.e., the indirect stakeholders.

4.5.1. Stakeholders, their knowledge and responsibilities

Though stakeholders of AmISs differ from domain to domain and given a domain from application to application, in health care the primary stakeholders are arguably always patients. Patients can be users of AmISs (e.g., in applications for assisted living), or indirect stakeholders as data subjects who do not interact with the system directly (e.g., in monitoring applications). In the case of smart homes for the elderly and people with physical disabilities, the group of patients can include non-disabled users, disabled users, elderly people, people with diminished vision, hearing impaired people, and cognitively impaired people [22]. The diversity of these types of patients makes it impossible to make general claims about the knowledge this group of stakeholders possesses. As a result, identification of (forward-looking) role responsibilities of this group of stakeholders is impossible. In the CCF2 case, the patients are the elderly residents of a care home that are not cognitively impaired. They are attributed a responsibility for their personal health (i.e., to look after their own health), although this responsibility is shared with the caregivers.

The second main group of stakeholders consists of the caregivers that can be sub-typed depending on the context of care and whether or not they are professional care givers. In a hospital or care home setting, caregivers include physicians of various specialisms, nurses (again in various types), and (non-professional) volunteers. As care shifts away from centralized, expert-driven models to one in personal living spaces in response to an ageing population, informal caregivers, such as family, friends, and members of the community are involved [23].

Professional caregivers can be expected to have the necessary knowledge and

skills for their profession and are expected to behave according to the responsibilities that come with their profession. In general terms for care giving professionals are expected to have the knowledge and skills to perform medical diagnosis (e.g., knowledge of anatomy, physiology, pathology, as well as operational knowledge about how to conduct diagnostic tests, how to interpret results of such tests) and/or the knowledge and skills how to treat what has been diagnosed (e.g., administer painkillers to a patient who is in pain).

Non-professional caregivers can be subdivided into those coming from the social network of the patient and unrelated volunteers. The basic difference between them is the reason why they have become caregivers. Family members and friends might take on care giving responsibilities because of social norms indicating the responsibility of taking care of family and those close to you [20]. Even so, out of a social network some people are more ready to take on these responsibilities than others. General volunteers might lack the social bond with the patient, but typically feel they contribute to society and the individual patients by offering their help.

The knowledge and skills for taking care of others of the group of non-professional caregivers is diverse. A young person, inexperienced in taking care of others, might not realize that she needs to check the temperature of a bath before placing someone in that bath. People that took care of their own small children can be expected to know about checking the temperature of the bath. As the background and intelligence of the large group of non-professional caregivers cannot be established upfront, only the general responsibilities that correspond to common sense can be attributed to the group as a whole. However, their individual background and intelligence might place additional individual responsibilities on them.

When introducing AmISs the knowledge on how to use the technology becomes relevant. Professionals are expected to make sure they know how to use the system, either through training or by studying its use themselves [23], depending on the use complexity of the system. However, who will make sure that non-professional caregivers that are allowed to use the system get the necessary training or instructions for properly usage? Fundamental to this chapter is the question: how do AmISs affect the knowledge condition of stakeholders and thus their responsibilities?

4.5.2. AmIS Architecture for Health Care Applications

AmISs are equipped with a set of sensors to gather data from the physical environment, and consist of intelligent agents to reason about that data, and to act upon that data to achieve the goals of the AmIS [19].

Sensors are devices that produce a measurable response to changes in their physical environment, e.g., temperature, humidity, movement, which electrical apparatus are in operation. Sensors can be arranged in wireless sensor networks (WSNs) to enable spatio-temporal sampling of physical, physiological, cognitive, and behavioral processes with a density that was previously impractical [23]. For health care monitoring, five basic subsystems have been identified [24]:

1. Body Area Network (WSN)

2. Personal Area Network (WSN)

3. Gateway to the Wide Area Networks

4. Wide Area Networks

5. End-user health care monitoring applications

Examples of components of Body Area Network (BAN) subsystems include RFID tags, electrocardiogram (ECG) sensors, and accelerometers worn by the patient [24]. Personal Area Networks (PAN) can consist of environmental sensors such as RFID readers, pressure sensors, luminosity sensors, and temperature sensors that gather contextual information about the person (or people) being sensed. Raw data of changes in one aspect of the environment, such as temperature, might not be useful in isolation. Therefore, data is fused. Data fusion is the process of putting together information obtained from many diverse sensors into a single representation of the environment [25]. These data can then be reasoned about by agents, e.g., by modeling the user, recognizing and predicting activity, and making decisions [19].

The gateway subsystem connects the BAN and PAN subsystems to wide area networks (WANs). The WANs comprise the infrastructure that distributes information from the BAN and PAN subsystems to the end-user applications. Agents can be employed at the sensor side, but also at the end-user application side. Finally, the collected information comes together in end-user applications (typically agents) that process and interpret collected data, and trigger actions. Additionally, (in the case of health care monitoring application) graphical user interfaces enable users to monitor vital signs in real time and receive alerts in case of emergencies [24].

Stefanov and colleagues [22] identify a number of types of home-installed devices for smart homes for the elderly and people with disabilities. Among those mentioned are devices for health monitoring and devices for information exchange. Devices in the former category include devices for monitoring vital signs, posture monitoring, behavior monitoring, recognition of facial expressions, and advanced chemical analysis. Devices in the latter category include systems for information access and telecommunication, systems for telemonitoring, teleinspection and remote control, and home network [22].

4.5.3. An AmIS for Independent Living for the Elderly
In the CCF2 project an AmIS was developed to support independent living for the elderly, and in particular to stimulate social and physical activity of elderly residents of a care home. The stakeholders identified for the system are described in Section 3.

Three scenarios were the point of departure: maintaining a social network, coaching, and intergenerational communication and engagement, see [3]. Support for these scenarios can be constructed from the basic pervasive health care monitoring applications identified in [24]: monitoring activities of daily living, fall and movement detection, location tracking, medication intake monitoring, and medical status monitoring.

In the first scenario, the system informs elderly residents of upcoming activities (such as group walks) and of which other residents intend to participate in these

activities, and allows them to share their preferences for participation with other users. By gathering sensor proximity information, the system can infer how frequently elderly residents have been near other residents, and hence how socially active they have been. The system also informs residents' caregivers of residents' participation in activities.

In the second scenario, agents in the system monitor residents' (levels of) physical activity and quality of sleep (inferred from the amount of movement registered by bed-mounted sensors). The agents notify the residents' caregivers if residents are not active enough, so that they can urge them to be more active. Personal coaching agents provide elderly residents with feedback, so that they can be urged to be more active or, if they are too active, they can be persuaded to reduce their level of activity. Furthermore, by aggregating information from all residents, the AmIS lets residents compare their health and physical activity with that of their peers.

The third scenario is about monitoring the communication between elderly people and their grandchildren. The system supports elderly residents in keeping track of their grandchildren's activities and vice versa, and in keeping each other up to date on daily achievements. The system encourages and enables elderly residents to remotely participate in their grandchildren's games.

To realize the functionality sketched in the scenarios the CCF2 AmIS was equipped with fixed and mobile sensor nodes. The fixed sensor nodes are deployed in fixed locations throughout a care home to form a wireless sensor network. For example, the resident's beds are fitted with accelerometers. The mobile sensor nodes worn by elderly residents contain accelerometers, photometers that measure the amount of light reaching the sensor, and a thermometer. Some patients wear a heartbeat sensor. Data from the sensors is used to derive all kinds of information. For example, data from the accelerometers can be used to determine the activity and mobility of the person wearing the sensor node. Combining this with data from the heartbeat sensor can provide information on how the amount a resident is moving affects her heart rate. The photometer and thermometer give general information about the wearer's environment that, in combination with other sensor data can be used to detect the activity of the wearer and even whether the user is indoors or outdoors (in case localization fails). The accelerometers in residents' beds measure how much the resident moves in bed which is an indicator of the resident's quality of sleep. The fixed sensor nodes can by cooperation determine the location of mobile sensor nodes. If multiple mobile sensor nodes are detected in each other's proximity, this provides an indication that the wearers of the mobile sensor nodes are engaged in a social activity. If there is a sudden acceleration detected by the accelerometer, and localization indicates that the mobile sensor is no longer changing location, this could indicate that the person wearing the mobile sensor node has fallen (or the mobile sensor node has fallen).

The CCF2 system contains an Activator agent that aims to persuade elderly residents to plan and participate in social activities [3]. The Activator agent notifies residents of activities that are going to take place and supports residents in self-monitoring their activities by providing them insight into how much physical and

social activity they have had in the recent past.

Information on upcoming activities the care home is organizing can be entered into an activity database by activity organizers. A resident database contains information about whether or not residents will participate. When residents accept or decline an activity, this gets registered in the resident database. The Activator pulls the number of people that are joining an activity from this database, and presents this to residents and activity organizers.

A Communal agent was developed to support activities in communal areas of the care home. Like the personal Activator, the Communal agent also displays upcoming activities, and measures the actual participation and activity of an ongoing event.

So, the CCF2 AmIS involves a combination of monitoring daily activities, detecting movement, and tracking location. These techniques are used for giving elderly residents of a care home insight into the social and physical activity levels of themselves and other residents. Also, professional caregivers as well as the social network consisting of some family members and friends can monitor activity levels of residents. In an abstract sense, the CCF2 AmIS aims at self-awareness (to persuade elderly users to adopt healthy behavior), and keeping professional and informal care networks aware. Hence, different types of knowledge are produced for different stakeholders.

4.5.4. Knowledge Production and Responsibilities in Ambient Intelligence

AmISs make various kinds of information available to users. Combined with the personal knowledge of their users, information can be deduced that is not deducible from the information produced by the system or users' knowledge alone.

In this section, we will examine how each stakeholder group we discussed above accesses information (made available by the underlying technology described above) and what knowledge each group gains based on the prior knowledge each group has. We will limit our discussion to the CCF2 case.

General practitioners
Pre-existing knowledge
General practitioners acquire much of their knowledge through their studies and in practice. This includes knowledge of standards, for example, the standards of the Dutch GP association (NHG) in the Netherlands.

Also, GPs must have knowledge of the International Classification of Primary Care (ICPC), provided by the World Health Organization [26]. The ICPC is a method for the classification of the treatment required for a patient presenting herself with a request for primary care. For this aim, the ICPC contains 17 chapters corresponding to different aspects of the human body, each divided into 7 components corresponding to different aspects of medical treatment.

Besides this general medical knowledge, GPs have knowledge of specific patients, consisting of a medical history and clinical signs. A medical history, is information gained through asking specific questions about demographics; chief complaint; history of present illness; past medical history; review of organ systems;

family diseases; childhood diseases; social history; regular and acute medications; allergies; sexual history. Clinical signs are gained through clinical examination. These can include vital signs such as temperature, blood pressure, and pulse.

Finally, GPs have knowledge of the responsibilities that come with their profession, which they gain from the Hippocratic oath and applicable codes of behavior and/or ethics.

Pre-existing responsibilities

The professional responsibilities of GPs consist of responsibilities derived from Hippocratic oath, including beneficence, non-maleficence and confidentiality. Also, GPs have responsibilities that are derived from codes of ethics, such as the World Medical Association's (WMA) International Code of Ethics, which includes responsibilities such as bearing in mind the obligation to respect human life.

Further, GPs have a responsibility to keep professional knowledge up to date. For example, in the Netherlands, GPs must take an exam twice a year testing their knowledge of the ICPC and of theoretical aspects of their profession.

New knowledge

Through use of the system, GPs gain detailed knowledge of patients' physical and social activity. This includes physiological data, location data, proximity to other people, and activities participated in. Before the introduction of the AmIS, GPs had to rely on patient's accounts of activities, complaints, and previous illness, and so on. That is, GPs had to take a medical history and rely on this to form a diagnosis and provide medical care. Additionally, GPs could perform tests (such as taking blood pressure) on site or order them elsewhere (e.g., blood tests).

The type of AmIS described here makes detailed information continuously available, providing a detailed picture of patients' condition over time. From the GPs point of view, it is as if the patient is undergoing continuous clinical examination and the GP can look into the resulting clinical signs at a given moment or over a longer period, gaining knowledge of trends and developments in a way that previously was not feasible.

Improved fulfillment of responsibilities

Due to the type of knowledge the system makes available, GPs are better able to diagnose and treat their patients, and to evaluate the outcomes of that treatment.

New responsibilities

The continuous availability clinical signals could lead to new responsibilities for GPs. Specifically, the availability of this information, combined with GPs' duty of beneficence, could lead to a responsibility to constantly act based on the information that is available to them. Not doing so could be considered a wrongful omission, as the GP could have known she was required to act. Fulfilling such a responsibility to act (at all times) on the continuous stream of clinical signs is not feasible, but this does raise an important question of who should monitor this information and who should act on it.

Nurses

Pre-existing knowledge

As a result of their training, nurses can be expected to have basic medical knowledge. Knowledge of care practice is part of this knowledge and includes:

- How to assess and diagnose needs

- How to plan outcomes, interventions

- How to implement interventions

- How to evaluate outcomes of care provided

- Knowledge of drug interactions

- Knowledge of specific patients' needs

- Knowledge of specific patients' care plans

More specifically, geriatric care nurses will have knowledge of special needs of elderly and afflictions particular to the elderly.

Pre-existing responsibilities

Nurses have a responsibility to assess and diagnose patient needs. Based on this, they have a further responsibility plan outcomes and interventions. They must implement interventions. Furthermore, they must evaluate outcomes of care. They also have a duty to inform patient of risks involved in various treatment options. As with other professional caregivers, it is essential that nurses keep their professional knowledge up to date.

New knowledge

The information made available by the system gives nurses detailed knowledge of patients' physical and social activity. Regarding physical activity, sensors continuously and ubiquitously collect information on location, amount (and speed) of movement, and pulse of the patient, as well as lighting conditions and temperature in the patient's environment. This basic information gives detailed knowledge of location, physical properties of location, and patients' physiological state through the course of the day (or any other time interval). Information on proximity to others and participation in social activities provides knowledge of levels of social activity.

Combined with the knowledge nurses already have of individual patients, their medical histories, their needs, and their care plans, knowledge emerges of care needs and outcomes (in different settings) at an entirely new level of detail and frequency.

Improved responsibility fulfillment

The knowledge that nurses can gain from the system makes it easier for them to fulfill their responsibility to assess the care needs of their patients. More specifically, by gaining insight into how every moment of the day is spent and how patients'

physiological condition changes throughout the day, nurses can better assess what kind of care is needed in certain circumstances. This knowledge also makes it easier for nurses to evaluate outcomes of the provided care.

New responsibilities
With new detailed continuous and ubiquitous knowledge of patients' activities and resulting detailed knowledge of care needs and care outcomes, it could be argued that a new responsibility arises to adjust care plans (and practice) whenever new information on needs and outcomes is available, which, with the introduction of the AmIS, is always. However, a nurse cannot realistically carry out the continuous adaptation of care plans for all of her patients. Hence, the introduction of the AmIS could potentially lead to a responsibility that nurses could not fulfill.

If a nurse were to know that a resident's behavior is unhealthy and the nurse, knowing that she should generally look after residents' health, decided not to act on this knowledge, then backward-looking responsibility for this wrongful act (omitting to look after the elderly resident's health) could fairly be attributed to the nurse. As for forward-looking responsibility, we argued that nurses have a general duty (or there is a norm) to look after elderly resident's health regardless of system use. Again, what the system changes, is the frequency with which the specific duty can arise. That is, whenever a nurse gains knowledge that a resident is acting in an unhealthy ways, a specific duty arises to ensure that that behavior is changed.

Family members
Pre-existing knowledge
Family members have intimate knowledge of elderly person, such as knowledge of the elderly person's identity and preferences. They can also be expected to have some form of knowledge of the elderly family member's basic physical condition, which they gain through direct observation during visits, from what their loved ones tell them, and from what staff members report.

Beyond knowledge of the elderly family member, family members with a caregiver role also have knowledge of "best care", that is, a vision of what criteria care should meet. As these family members monitor care provision to their loved ones (through direct observation and querying elderly loved one), they will have some knowledge of how care is conducted. Finally, family members will have some knowledge of responsibilities of care home staff and of their own responsibilities.

Pre-existing responsibilities
One of the primary responsibilities family members see themselves as having is overseeing the care of the loved one, representing the resident's perspective and history, and keeping the family connections [27]. Further responsibilities perceived by family members include:

- maintaining continuity (based on relative's intimate knowledge of the older person), which involves helping the older person to maintain their sense of identity through the continuation of loving family relationships and through helping staff to get to know the resident as an individual;

- keeping an eye, by monitoring the care received, providing feedback to staff and filling any gaps; and

- contributing to community through interacting with other residents, relatives and staff, taking part in social events and generally providing a link with the Outside world [28].

New knowledge
With the introduction of an AmIS, family members gain knowledge of how their loved ones spend their days in terms of physical and social activities. This goes beyond the knowledge family members normally gain from direct observation in visits, from what their loved ones tell them, and from what staff reports. It is different in that it consists of all activities engaged in throughout the day, with additional knowledge of their loved one's specific locations, physical properties of those locations (e.g., temperature), and their loved ones' physiological responses to those activities and environments. This knowledge then gives some insight into the effects of the care the loved one is receiving, though family members will not necessarily have the knowledge required to interpret these effects.

Improved fulfillment of existing responsibilities
The new knowledge for family members that the AmIS produces improves family members' ability to carry out their duty perceived duty of keeping an eye, in that it allows them to scrutinize in greater detail and with greater frequency how care is being provided.

New responsibilities
With the introduction of the AmIS, family members potentially have knowledge of their elderly family members' activities, levels of activity, and physical condition at all times. In light of their responsibility to keep an eye on the care received family members might feel a needs to interpret the knowledge available to them in order to scrutinize care. In light of their perceived responsibility to maintain continuity by helping staff get to know the elderly family member as an individual and in light of their perceived responsibility to represent the elderly person's perspective, they might feel the need to explain the elderly person's actions and responses to activities to staff. Considering the continuous availability of information on activities and the like, such explanations could burden the family members giving them, as well interfere with caregivers' work.

Elderly residents
Pre-existing knowledge
Elderly residents of care homes can be expected to have knowledge of their own physical condition. The level of detail of this knowledge will vary. The most important source of this knowledge is introspection – people know something about their own physical condition because they know how they feel. A further source of this knowledge is caregivers. For example, an elderly person's GP might have told her she has diabetes. In that case, she will know she has diabetes (though, as a layperson, she might not understand the full implications of that diagnosis).

Furthermore, assuming they are not (significantly) cognitively impaired, the elderly can be expected to have knowledge of their own activities of daily life and other activities they engage in. For example, on might know (based on recollection) that this morning she woke up, had breakfast alone in her room, went downstairs to play chess, and went up to take a nap afterwards, from which she has just awakened. Finally, the elderly should have some knowledge of what behavior is good for their health and what behavior is bad for their health. For example, an elderly person should be aware that going out in the cold without warm clothes could lead to a cold.

Pre-existing responsibilities
One primary (though debatable) responsibility for the elderly (or for any person with adequate capacities) is the individual responsibility for their own health. That is, elderly people should, to some extent, do things that are good for their health and avoid those things that are not. As residents of a care home, elderly people also have a responsibility to respect other residents and staff, and follow facility rules.

New knowledge
Information collected from sensors is presented to elderly residents through the performance state (or view) of the Activator interface. This gives them insight into how physically and socially active they have been. It also allows them to compare this to the average levels of the other users.

The combination with their prior knowledge of which activities they participated in (and with whom), how physically active they were, and how much activity is healthy for them, allows users to gain more detailed information on their levels activity than could be gained from introspection alone. For example, they could come to know how their levels of activity affected their heartbeat (though we can not expect them to have the knowledge required to properly interpret information such as heartbeat). Furthermore, if the system shows that their activity levels have been higher or lower than the elderly residents know they should be, then they come to know that their level of activity has not been healthy and that they should be more or less active.

Improved fulfillment of existing responsibilities
Feedback on how activities influence pulse and health helps elderly better look after their health by adopting healthy levels of activity.

New responsibilities
In the CCF2 case, elderly users gain detailed knowledge about their social and physical performance. This is not just knowledge of their actual activity levels, but also knowledge of how their activity levels compare to some standard. That is, they gain knowledge that they are not as active as or more active than they should be. By knowing this, and deciding not to change their behavior, we could say they are responsible for the state of their (social and/or physical) well-being. If the

other conditions for responsibility are met, these stakeholders' use of the system could contribute to (backward-looking) responsibility being ascribed to this group of stakeholders. Also, gaining knowledge that they are not active enough or too active could give them a forward-looking responsibility to be more or less active.

Stakeholder	Knowledge	Responsibilities
General Practitioner	Standards	Hippocratic Oath (Forward-looking)
	Classification of primary care encounters	Codes of Ethics (Forward-looking)
	Medical history of specific patient	Keep professional knowledge up to date (Forward-looking)
	Clinical signs of specific patient	Monitor information (New, forward-looking)
	Knowledge of professional responsibilities (Hippocratic Oath and applicable codes of ethics)	Constantly act on information (New, forward-looking)
	Specific patient's physical activity (New)	
	Specific patient's social activity (New)	
	Environmental conditions (New)	
	Trends (New)	
Nurse	How to assess and diagnose needs	Assess and diagnose patient needs (Forward-looking)
	How to plan outcomes and interventions	Plan outcomes and interventions (Forward-looking)
	How to implement interventions	Implement interventions (Forward-looking)
	How to evaluate outcomes of care	Evaluate outcomes of care (Forward-looking)
	Knowledge of drug interactions	Inform patient of risks involved in treatment options (Forward-looking)
	Knowledge of specific patients' needs	Keep professional knowledge up to date (Forward-looking)
	Knowledge of specific patients' care plans	Continuously adjust patients' care plans (New, forward-looking)
	Knowledge of special needs of elderly	Ensure unhealthy behavior is changed (New, forward-looking)
	Knowledge of afflictions particular to the elderly	Monitor information (New, forward-looking)
	Specific patients' physical activity (new)	

4

	Specific patients' social activity (new)	
	Current location of patients	
	Current physiological state of patients	
	Level of social activity	
Family members	Elderly person's identity and preferences	Overseeing care of loved one (Forward-looking)
	Elderly person's basic physical condition	Representing loved one's perspective and history (Forward-looking)
	Knowledge of "best care" for family member	Keeping family connections (Forward-looking)
	How care is conducted	Maintaining continuity (Forward-looking)
	Responsibilities of care homes staff	Keeping an eye (Forward-looking)
	Own responsibilities	Contributing to community
	Physical activity (including location, physical properties of locations, physiological information (New)	Interpret available information to scrutinize care (New, forward-looking)
		Explain elder's actions to care professionals (New, forward-looking)
Elderly	Own physical condition Own activities of daily living Other activities Knowledge of what behavior is good for health Detailed physical activity (new) Detailed social activity (new) Average level of activity of others (new) Effect of behavior (new)	Taking care of own health (Forward-looking) Respect other residents and staff State of (social and/or physical) well-being (New, Forward-looking/Backward-looking) Be more or less (socially or physically) active (New, forward-looking)

Table 4.1: Summary of stakeholders, their knowledge and responsibilities

4.6. The Moral Implications for AmIS Developers

In the previous section we argued that the use of AmIS leads to certain knowledge in users, which, through the knowledge condition of responsibility, leads to responsibilities for those users. So, by using the type of system described here,

people could gain duties and could be praised or blamed from their actions or omissions. So, these systems could affect people's lives in important ways. We argue that designers of such systems should take this into account when designing such systems. In this section, we offer initial guidelines on how to do so.

One of the main issues with the way responsibilities arise through the use of these systems is that there is a possibility that users do not realize that responsibilities arise from using the system. We cannot expect all users to know that by using these systems they gain specific types of knowledge which, if the users act in certain ways, could contribute to them being blamed (or praised) for the outcome of their acts. Similarly, we cannot expect all users to know that by having access to certain information, they acquire certain duties. As users cannot be expected to foresee these implications of using these systems, designers should at the very least ensure that users understand, or are given the possibility of understanding, the implications of being a user.

Informing the user of the implications of use could be taken a step further by requiring users' informed consent in order for them to use the system. A problem with this is that giving the user only two options, consent (with taking on responsibilities) and use, or refrain from use altogether approaches blackmailing [29]. Especially in the case of the type of system described here, which is made part of the infrastructure of the care home and used for health care, this all-or-nothing approach is undesirable. A better approach would be to offer users insight into the implications of using particular parts of the system or accessing particular kinds of information and letting them make (informed) decisions about whether or not to use each function based on that insight. This would involve explaining the implications of each function to users.

Discussion so far has been about informing users so that they can decide whether or not they want to take on the responsibilities that come with certain types of knowledge that can be acquired by using the system. This does not take into account the stakeholders about whom the system provides information, that is, the stakeholders whom the system monitors. These stakeholders could take issue with the fact that by being monitored by the system, they are making monitoring information available to family members and others, and in doing so they are in a sense imposing responsibilities on these family members and others. As with other users of the system, these 'data subjects' should be given the opportunity to understand what responsibilities they are creating by being monitored. Further, they should be given the ability to choose whether or not they want to make available the different kinds of information that can lead to responsibilities, and they should be able to do so in an informed way.

For designers to be able to inform users how use of their systems can lead to responsibilities, designers must first understand how use of the system leads to responsibilities. To do so, designers should conduct (or have access to) stakeholder analyses. They should find out and be aware of who the direct and indirect stakeholders of the envisioned system are. Furthermore, they should analyze which responsibilities these stakeholders have prior to using the envisioned system, which knowledge these stakeholders can be expected to have, and which knowledge use

of the system results in for each of these stakeholder groups. The AmIS should support the users in fulfilling their responsibilities. Ensuring that they do belongs to the responsibilities of the AmIS developer. For example, the developer could build in artificial agents into the AmIS, supporting the responsibilities of the human users by alerting them, preventing and/or reporting misuse of the system, alerting officials or care organizations, refraining from observing when not necessary for the purpose of the system, etc. We consider that the idea of artificial agents supporting ethical behavior (by supporting human users' in fulfilling their responsibilities) falls within the realm of machine ethics, which, in one sense, deals with questions of whether ethics can be "put into" a machine [30]. One way of doing so, according to Moor, would be to constrain the machine's actions to avoid unethical outcomes, making it an implicit ethical agent [30]. According to Moor, a stronger sense of moral agency could be achieved by creating explicit ethical agents that can represent ethical categories and perform analysis. One example of work in this direction is Deontic Epistemic Action Logic (DEAL), which models statements on permission and obligation, beliefs and knowledge, and actions [31, 32]. A final type of ethical agent that Moor describes is the full ethical agent, although it is debatable whether a machine can be one. An explicit ethical agent approach such as DEAL could help realize systems that support users in fulfilling their responsibilities, by having the analyzing (and reasoning about) responsibilities and acting appropriately in light of these.

More generally, we conclude that the responsibility of the AmIS-developer to design for responsible use of his system, involves at least the following criteria: 1) Developers need to be aware of foreseeable responsibilities of the usage of their system (intended or unintended). 2) The system needs to prepare intended users for their moral responsibilities before the user is accepted as a user. 3) The intended user needs to accept these responsibilities when accepting to use the system. 4) With respect to professional usage and anticipated responsibilities for non-professionals the system and its agents should have a regular reminder of moral responsibilities.

4.7. Conclusion

In this chapter, we aimed to address the responsibilities associated with Ambient Intelligence. Central to our arguments was the fact that knowledge leads to responsibilities. Because Ambient Intelligence Systems (AmISs) provide their users with information and support the users in using that information to extend their knowledge, the use of AmISs leads to responsibilities for both intended and unintended users. By making developers aware of this, through the current chapter, a reflective argument leads to the conclusion that developers of AmISs have the responsibility to design for the responsibilities of the users of theirs systems. We have formulated an initial list of criteria for how the developer can take up this responsibility.

Many questions regarding responsibility in the use and development of AmISs deserve further investigation. For example: What is the relation (or what should the relation be) between our conclusions and legal obligations, for the developer,

for the users, for other stakeholders? Can we attribute responsibility to artificial agents? Would a "license to use" agreement be an adequate measure? And on a higher level: does the access to a tool that has certain information (regarding someone's health) bring or imply the moral obligation to use the tool? We hope this chapter contributes to putting such questions on the agenda of the AmIS research community.

References

[1] M. J. Zimmerman, *Moral responsibility and ignorance,* Ethics **107**, 957 (1997).

[2] R. Davis, *Knowledge-based systems,* Science **231**, 957 (1986).

[3] N. Romero, J. Sturm, T. Bekker, L. de Valk, and S. Kruitwagen, *Playful persuasion to support older adults‚äô social and physical activities,* Interacting with Computers **22**, 485 (2010), <ce:title>Special Issue on Inclusion and Interaction: Designing Interaction for Inclusive Populations</ce:title>.

[4] M. Wooldridge and N. R. Jennings, *Intelligent agents: Theory and practice,* The Knowledge Engineering Review **10**, 115 (1995).

[5] N. A. Vincent, *A structured taxonomy of responsibility concepts,* in *Moral Responsibility: Beyond Free Will and Determinism*, edited by N. A. Vincent, I. Van De Poel, and M. J. Van Den Hoven (Springer, 2011).

[6] H. L. A. Hart, *Punishment and responsibility,* (Oxford: Clarendon Press, 1968) Chap. IX. Postcript: Responsibility and Retribution.

[7] K. G. Coleman, *Computing and moral responsibility,* in *The Stanford Encyclopedia of Philosophy*, edited by E. N. Zalta (2008) winter 2008 ed.

[8] H. Nissenbaum, *Accountability in a computerized society,* Science and Engineering Ethics **2**, 25 (1996).

[9] N. Doorn, *Responsibility ascriptions in technology development and engineering,* Science and Engineering Ethics , 1 (2009).

[10] T. De Lima, L. Royakkers, and F. Dignum, *A logic for reasoning about responsibility,* Logic Journal of the IGPL **18**, 99 (2010).

[11] N. Doorn and I. van de Poel, *Editors' overview: Moral responsibility in technology and engineering,* Science and Engineering Ethics , 1 (2011).

[12] R. Mastop, *Characterising responsibility in organisational structures: The problem of many hands,* in *Deontic Logic in Computer Science*, edited by G. Governatori and G. Sartor (Heidelberg: Springer, 2010).

[13] G. Sher, *Who Knew?: Responsibility without Awareness* (Oxford University Press, 2009).

[14] M. J. Zimmerman, *Negligence and moral responsibility,* Nous **20**, 199 (1986).

[15] M. S. Moore and H. M. Hurd, *Punishing the awkward, the stupid, the weak, and the selfish: The culpability of negligence,* Criminal Law and Philosophy **5**, 147 (2011).

[16] J. L. Austin, *How to Do Things with Words*, 2nd ed., edited by J. O. Urmson and M. Sbisá (Harvard University Press, 1962).

[17] J. Searle, *Austin on locutionary and illocutionary acts,* The Philosophical Review **77**, 405 (1968).

[18] J. Searle, *Speech Acts: An Essay in the Philosophy of Language* (Cambridge University Press, 1969).

[19] V. R. J. Diane J. Cook, Juan C. Augusto, *Ambient intelligence: Technologies, applications, and opportunities,* Pervasive and Mobile Computing , 277 (2009).

[20] S. H. Matthews and T. Tarler Rosner, *Shared filial responsibility: The family as the primary caregiver,* Journal of Marriage and Family **50**, 185 (1988).

[21] M. Weiser, *The computer for the 21st century,* Scientific American **265**, 94 (1991).

[22] D. H. Stefanov, B. Zeungnam, and W.-C. Bang, *The smart house for older persons nad persons with physical disabilities: Structure, technology arrangements, and perspectives,* IEEE Transactions on Neural Systems and Rehabilitation Engineering **12**, 228 (2004).

[23] J. Ko, C. Lu, M. B. Srivastava, J. A. Stankovic, A. Terzis, and M. Welsh, *Wireless sensor networks for healthcare,* in *Proceedings of the IEEE*, Vol. 98 (2010) pp. 1940–1960.

[24] H. Alemdar and C. Ersoy, *Wireless sensor networks for healthcare: A survey,* Computer Networks **54**, 2688 (2010).

[25] H. Lee, K. Park, B. Lee, J. Choi, and R. Elmasri, *Issues in data fusion for healthcare moitoring,* in *Proceedings of the 1st International Conference on Pervasive Technologies Related to Assistive Environments* (2008) pp. 1–8.

[26] W. H. Organization, *International classification of primary care, second edition,* (2011).

[27] M. Bern-Klug and S. Forbes-Thompson, *Family members' responsibilities to nursing home residents:" she is the only mother i got".* Journal of gerontological nursing **34**, 43 (2008).

[28] S. Davies and M. Nolan, *'making it better': Self-perceived roles of family caregivers of older people living in care homes: A qualitative study,* International journal of nursing studies **43**, 281 (2006).

[29] M. Langheinrich, *Privacy by design - principles of privacy-aware ubiquitous systems,* in *Ubicomp 2001: Ubiquitous Computing*, edited by G. D. Abowd, B. Brumitt, and S. Shafer (Springer, 2001) pp. 273–291.

[30] J. H. Moor, *The nature, importance and difficulty of machine ethics,* IEEE Intelligent Systems **21**, 18 (2006).

[31] J. van den Hoven and G.-J. Lokhorst, *Deontic logic and computer-supported computer ethics,* in *Cyberphilosophy: The Intersection of Computing and Philosophy*, edited by J. H. Moor and T. W. Bynum (Blackwell, 2002) pp. 280–289.

[32] V. Wiegel, M. J. Van Den Hoven, and G.-J. Lokhorst, *Privacy, deontic epistemic action logic and software agents: An executable approach to modeling moral constraints in complex informational relationships,* Ethics and Information Technology **7**, 251 (2005).

4

5

Putting Value(s) into Requirements Engineering

Software systems can give rise to ethical issues, with human values such as privacy, autonomy and responsibility at their heart. Such issues often do not become apparent until software has been put to use, and are left to be dealt with after harm has been done. However, these ethical issues are influenced by decisions made during design. A range of approaches to technology design aims to consider ethical issues and their underlying values during design, when there is more room to shape a technology. These values-oriented approaches guide designers in identifying and analyzing value issues with technology, but do not focus on deriving requirements from identified issues. As a result, the knowledge that can be gained from values-oriented methods does not find its way into requirements engineering processes. To address these challenges, we present a method to extract values and related elements obtained through existing value elicitation techniques, and to document these elements in a format that is amenable to use in further requirements engineering activities. We present a case study as a proof of concept.

5.1. Introduction

Software affects the social and physical context into which it is deployed by introducing new capabilities or modifying existing ones, and by changing the way people do things. Such effects often have ethical implications that center on a system's impact on human values. Privacy and accountability issues with electronic patient record systems, bias in search engines, and user autonomy issues in decision support systems are but a few examples of ethical implications resulting from values being hindered.

Technology's impact on values is often not considered until the technology has been put to use. To an important extent, this impact is the result of the technology's functionality and qualities. These are, in turn, the result of decisions made during the design process. That is, design decisions strongly influence how a software system eventually impacts human values. This suggests that technology's impact on values need not be an afterthought; rather, values can be considered during design, when technology is still malleable.

Values are receiving increasing attention in technology design disciplines. The field of Human-Computer Interaction and related disciplines have seen a considerable amount of work on values over the past 20 years. Methods that deal with values explicitly have emerged and matured, perhaps most prominently within the Value Sensitive Design (VSD) framework [1]. These methods guide designers in identifying, conceptualizing, and analyzing values in light of a technology, evaluating technology, and pro-actively designing technology to support values.

As many of these methods emerged within Human-Computer Interaction, they were not specifically developed for use in requirements engineering, and the products of such methods are not readily usable for use in requirements engineering activities. Methods developed in VSD and related approaches can help identify value issues, but they do not offer guidance on translating values to requirements. In Requirements Engineering itself, there are few approaches that address values and value issues, but these methods have not been tested and tried to the extent that more mature methods in VSD have. As a result, even though values are elicited, they potentially do not find their way into requirements, or are not considered to the extent they could be in design.

To address these issues, we present an approach to tackle the challenge of integrating values-oriented techniques into a requirements engineering process. Our approach supports requirements engineers and stakeholders in identifying values and translating them into a form that is suitable for use in the further requirements engineering process.

In Section 2, we will discuss related work in Value Sensitive Design and Requirements Engineering. In section 3 we will present our approach. We will illustrate this approach in a case study in Section 4, and discuss the approach and draw some conclusions in Section 5.

5.2. Related work

In this section, we discuss key concepts in Value Sensitive Design, the most prominent values-oriented framework to have emerged within Human-Computer Interaction. Furthermore, we discuss approaches in Requirements Engineering that deal with values to some extent.

5.2.1. Value Sensitive Design

Value Sensitive Design (VSD) [1] draws on a number of key concepts. It distinguishes between *direct stakeholders*, who directly interact with technology, and *indirect stakeholders* who are affected by the technology or its output without interacting with it. *Values* are another key concept, which work on VSD has defined as "what a person or group of people considers important in life" [1]. VSD adopts an interactional stance towards values, which considers values neither solely determined by technology nor by social systems, but by a combination of the two. VSD distinguishes between three types of values. *Explicitly supported values* are those that are required to be supported by the technology. *Designer values* refer to the designer's or researcher's personal values that implicitly guide design decisions. *Stakeholder values* are values that are important to some, though not necessarily all, stakeholders of a technology [2]. Differences among values can give rise to *value tensions* (such as privacy versus security) among stakeholder groups, when supporting one value in a technology challenges another.

These concepts are central to VSD's three-part methodology, consisting of conceptual investigations of key stakeholders and values; empirical investigations of actual or potential stakeholders and contexts-of-use; and technical investigations aimed at designing new technology to support selected values or examining how existing technologies support or hinder certain values [3]. These investigations can be applied iteratively and integratively throughout the design process.

Several specific methods have been developed within VSD, some of which are suited to elicit knowledge about values *in situ*. The Value Scenarios technique supports envisioning long-term systemic effects of new technology, based on scenario-based design and design noire [4]. Designers or stakeholders write these scenarios and consider stakeholders, pervasiveness, time, systemic effects and value implications of the technology.

The Value Dams and Flows method aims to translate results of stakeholder surveys into specific, implementable design features, in a way that was systematic and empirically informed [5]. The method accounts for values in design by avoiding features that are particularly problematic to stakeholders, by identifying and designing for values that stakeholders do wish to see the system embody, and by systematically addressing design tradeoffs that concern values.

Envisioning Cards [6] are a design toolkit that supports designers in considering potential ethical implications of new technologies. The cards aim to raise awareness of long-term and systemic effects of proposed technology by guiding the design process through explanations of key themes and associated concepts, and providing focused design activities to address issues related to these themes in a systematic way. The toolkit consists of cards that fall into one of four categories of envisioning

5

criteria: stakeholders, time, values, and pervasiveness.

5.2.2. Requirements Engineering

A variety of elicitation techniques is available, which differ depending on the effort required and type of information they help elicit. Zowghi and Coulin [7] identify eight core techniques that cover the spectrum of available techniques: interviews, domain analysis, groupwork, ethnography, prototyping, goal-based approaches, scenarios, and viewpoints (echoing the classes of elicitation techniques identified in [8]). Although these techniques cover a broad range, few focus explicitly on ethical implications or stakeholder values.

Notable exceptions include [9, 10]. Thew and Sutcliffe propose a technique that that uses a taxonomy of users' values (principles and norms users associate with their work role), motivations and emotions, and process guidance to elicit and analyze these issues. The taxonomy of values includes nine high-level categories of values, listing related terms (or subcategories) for each in order [9]. Each category also includes potential sources or conversation topics to help uncover values. Finally, process implications are listed for each value category. The taxonomy can guide preliminary analysis, helping identify key issues (values that are potentially relevant). By making these explicit, the method encourages gathering evidence to support initial 'hunches' using interviews, ethnographic techniques, storyboards and prototypes.

[10] present a technique focuses on approximating or identifying users' values based on their preferences for key (work) tasks. In this method, the researcher first identifies key tasks that are potentially value-dependent. Next, the researcher determines which tasks can actually predict dominant values, by having a sample of users answer a questionnaire to indicate their preferences for previously identified potentially value-dependent tasks. This results in the identification of key task-value relationships. Users complete another questionnaire to indicate their preferences for each of the key tasks. User preferences for several key tasks related to the same value are used to approximate the user's preference for that value.

These techniques help draw attention to values in the requirements engineering process. They do so by developing their own means of eliciting values. While these are important contributions, we argue that it is a missed opportunity not to use techniques developed in the last 20 years of work on VSD. In the next section, we propose an approach to bridge the gap between VSD and requirements engineering.

5.3. Our proposal: from values to requirements

In this section, we propose a technique to elicit requirements of a system to be developed that is based on a value analysis of the system's stakeholders. By that, we aim to connect the concepts used in value sensitive design (VSD) to those used in requirements engineering (RE), and provide researchers in both fields the opportunity to profit from each other's work. The technique involves the following five steps.

1. Analyzing the system's stakeholders

2. Analyzing the stakeholders' values

3. Providing concrete situations with the values

4. Determining stakeholder needs

5. Creating user stories

In the remainder of this section, we will explain each step, and give a suggestion for a practical use of the technique.

The first step, a stakeholder analysis, involves the identification of the direct and indirect stakeholders of the system at stake. The second step, a value analysis, involves the identification of values that are relevant for the stakeholders (in that role) identified in the first step. Stakeholders and values are core concepts of the VSD methodology, and the first two steps of the proposed technique are often used within VSD techniques. For instance, the value elicitation techniques 'envisioning cards' and 'dams and flows' described in Section 2.1 also involve the analysis of stakeholders and their values. The envisioning cards deck contains cards with the assignments like "create a list of the system's direct stakeholders", "generate a list of 3-5 indirect stakeholders", "generate as list of as many potentially implicated values as possible", and "elicit stakeholder views and values". And the dams and flows technique involves identifying the stakeholders of the envisioned technology, identifying potential harms and benefits of the system, and then analyzing values underlying the harms and benefits.

The third step of our technique, concrete situations, involves coming up with one or more concrete situation(s) that explain(s) why or how a value is important for a stakeholder. This is important because one value can have multiple meanings for a stakeholder. For example, the value of privacy for a user of a system can mean that the user's personal information stored in the system should not be shared with anyone, or that the user should be able to use the system on his own.

Providing one or more concrete situations associated with a stakeholder's value is a preparation for the fourth step, stakeholder needs. In this step, it is determined how the stakeholder's value can be supported in that concrete situation. For instance, to support (protect) a user's privacy with regard to undesired sharing of personal information, the user should be able to indicate which personal information can be shared with whom and under what conditions.

A notion of stakeholder need is already closely related to the concept of a requirement. To fully connect our approach to RE, however, we added a fifth step, the creation of user stories. User stories are commonly used in RE to capture system requirements, and there they are usually of the form: "As a [role] I want [something] so that [benefit]". We propose to create user stories according to the following template: "As a [stakeholder] I want [stakeholder need] so that my [value] is promoted/supported when [concrete situation]". Note that one stakeholder can have multiple values, one value can have multiple concrete situations, and one concrete situation can have multiple stakeholder needs, but that one user story is created for each stakeholder need (with the associated stakeholder, value and concrete situation). This last step of our technique creates a result that is based on techniques

and concepts from the VSD field, and that is known and can be dealt with in the field of RE.

One way to use this technique in practice is to organize a workshop with stakeholders of the envisioned system, e.g. domain experts, potential users, and developers. We suggest the following outline for such a workshop.

A. Short presentation to introduce the participants to values and VSD, e.g. by providing examples of values in design (assuming that the participants are not familiar with VSD yet)

B. Identify direct and indirect stakeholders

C. Per stakeholder (this may be a selection of stakeholders identified in step B), identify one or more values and concrete situations

D. Per concrete situation, identify one or more stakeholder needs

The identification of stakeholder values and provision of concrete situations are taken together in one part (part C) because in a workshop setting it is natural to immediately explain a value to the other participants by providing a concrete situation. In a large groups (e.g., when $n > 8$), part C and D can be done in subgroups of 4 to 5 participants. In that case, we suggest to perform part C and D together with all participants for the first stakeholder so that everybody understands what is expected of them. The last step of the technique, creating user stories, does not require new input of the participants, and can thus be performed by the workshop organizers afterwards.

To conclude, we proposed a technique to elicit requirements while accounting for values. The technique connects important concepts in the field of VSD (stakeholders and values) to a concept widely used in RE (user stories). The field of RE provides techniques and tools to incorporate the requirements phrased in user stories in the system design process. The proposed technique has two benefits as compared to other requirement elicitation techniques. First, paying explicit attention to values of direct and indirect stakeholder may lead to requirements that would not have been discovered otherwise. Second, knowing the values behind requirements provides an extra perspective in case of design trade-offs. This perspective allows developers to also reason and decide about underlying values instead of mere system features.

5.4. Case study

In this section we describe a case study in which we tested the technique proposed in the previous section by organizing a value-requirements workshop. First, we will introduce the project in the context of which we performed the workshop, then we will describe the workshop results, and based on that, we provide an evaluation of the proposed technique.

5.4.1. Context

The case study was performed in the context of the IQmulus project[2]. This is a 4-year European project in the area of intelligent information management. The main objective of the project is to enhance decision making by developing a system (the IQmulus system) that extracts relevant information from large, heterogeneous geo-spatial data sets. Such a system is needed because new data acquisition techniques are providing fast and efficient means for multidimensional spatial data collection, e.g. stereophotogrammetry, airborne LIDAR surveys, and SAR satellites. These techniques provide extremely high volumes of raw data, but in order to be useful, the heterogeneous data sets require harmonization and integration. The IQmulus project aims to make these data more accessible, and by that, help decision makers to make better choices, e.g. in case of flooding, flash floods or industrial accidents.

The project consortium involves partners with technical expertise, e.g. in the development of algorithms for data integration and information filtering, the visualization of data, or the development of architectures, and partners with domain expertise, e.g. collecting and using geo-spatial data for marine spatial planning or for rapid response and territorial management on land. The latter group are potential future users of the IQmulus system. The case study was performed in year 1 of the project. At that point, several other user workshops and requirement elicitation activities had been organized.

5.4.2. Value Stories Workshop

The value-requirement workshop was performed with representatives of several of the IQmulus consortium partners with domain expertise, i.e., some of the potential users of the IQmulus system. More specifically, the workshop involved 9 participants with representatives of the following institutes: 2 of IGN (French National Institute of Geographical Information and Forestry), 2 of Fomi (Hungarian Institute of Geodesy, Cartography and Remote Sensing), 2 of Regione Liguria (Genova, Italy), 1 of Ifremer (French Institute for Exploitation of the Sea), 1 of UBO (European Institute for Marine Studies), 1 of HR Wallington (Independent Research and Consultancy in Civil Engineering and Environmental Hydraulics). The workshop was performed such as described in Section 3, and led by the authors of this paper. The total duration of the workshop was 4 hours (excluding breaks). None of the participants was familiar with VSD before participation in the workshop. In the remainder of Section 4.2 the results of the workshop are provided.

In the first step, stakeholder analysis, the following 13 direct and indirect stakeholders of the IQmulus system were identified: consortium partners, public, European Committee, and users, which were divided into managers, GIS experts, water authorities, scientists, disaster management people, risk assessment people, spatial planners, data providers, operators (hardware and infrastructure), and decision makers. The participants agreed that the public and European Committee are indirect stakeholders, but that the other stakeholders can be either direct or indirect stakeholders of the IQmulus system.

---

[2]http://www.iqmulus.eu

From this list, three stakeholders were selected for further analysis. It was tried to select three stakeholders that are very distinct, but yet influenced by each other. The following selection was proposed by the workshop leaders, and agreed upon by the participants.

Decision makers who do not directly interact with the system (indirect stakeholder)

GIS experts who directly interact with the system in order to provide information to decision makers (direct stakeholder)

Residents of an area with flood risk, representing 'public' (indirect stakeholder)

For these three stakeholders 31 values were identified, 50 concrete situations and 94 stakeholder needs. This resulted in 94 user stories. Due to space limitations, we cannot provide all these results here. Instead, we selected two representative user stories for each stakeholder. Table 5.1 shows the user stories with the associated stakeholders, values, concrete situations, and stakeholder needs, respectively.

Several interesting observations can be made from the table. The decision maker's value of personal job security can lead to two conflicting stakeholder needs. On the one hand, personal job security may yield a need for metrics about the uncertainty of information (user story 1), but on the other hand, it may yield the need to not receive these metrics (user story 2). In the former case, the decision maker takes responsibility for the interpretation of (processed) data, and in the latter case, GIS experts take this responsibility. This tension is related to the GIS expert's value of accountability. The more responsibility is shifted from the decision maker to GIS experts, the more important their need for information about data and algorithms becomes to account for the information they delivered to the decision maker (user story 3).

The table also shows a relation between the values of residents and those of the decision maker. For residents of a flood area, having information about the flood risks can increase residents' trust in the information provided by decision makers (user story 5). This is in tension with user story 2, according to which the decision maker wants information of the type 'yes' or 'no' and 'safe' or 'unsafe', instead of information about risk sizes.

User stories 1, 2, 3 and 5 are all related to the question of how much information the IQmulus system should provide about risks, uncertainty, and accuracy of the data it produces. The user stories show the impact of design choices regarding that question on the values of all three stakeholders. User story 4 and 6 are not direct requirements of the IQmulus system. However, the design of the IQmulus system may be impacted by the stakeholder needs in these user stories. For example, the IQmulus system could have a decision support function that selects central, but safe locations from where voluntary actions could be coordinated.

5.4.3. Observations

The workshop participants generally indicated that they liked the workshop, that they found it interesting to learn about VSD, and that they thought that VSD pro-

Stake-holder	Value	Concrete situation	Stakeholder need	User story
Decision maker	Personal job security	Being fired because of a wrong decision	information should be accompanied by metrics about its (un)certainty and/or accuracy if possible (this may increase responsibility of DM)	1. As a decision maker I want information with metrics about its (un)certainty and/or accuracy if possible to support my personal job security
			avoid fuzziness in information, provide 'yes' or 'no' type of information (this may increase responsibility of expert)	2. As a decision maker I want information without fuzziness to support my personal job security
GIS expert	Accountability	expert has to rely on processing, and tools when accounting for data provided	information about the quality of input data, quality of algorithms, knowledge of expert, and rightness of choices made	3. As a GIS expert I want information about the quality of input data, quality of algorithms, knowledge of expert, and rightness of choices made to support my accountability for data provided
		Awareness of accuracy of data in order to interpret results correctly	Training so that GIS experts know what is possible and why, which algorithm does what	4. As a GIS expert I want training to know what is possible and why, and which algorithm does what to support my accountability
Resident of an area with flood risk	Trust	believing that decision makers, authorities, system will provide correct information	information about potential risks	5. As a resident of an area with flood risk I want information about potential risks to support my trust in information provided by decision makers
	Solidarity	Having a way to help in crisis, play a part	information, coordination, and distribution of tasks depending on each person's expertise	6. As a resident of an area with flood risk I want a mechanism that coordinates and distributes tasks depending on each person's expertise to support solidarity

Figure 5.1: Identified stakeholders and values

vided a valuable perspective on the use and development of the IQmulus system, a perspective they had not encountered before. The participants particularly mentioned that it helped them to reflect on the main aims of the project. Some of the workshop participants had been involved in organizing user workshops to obtain requirements for the IQmulus system themselves. They remarked that the experience and knowledge obtained in these user workshops helped them a lot in the value-requirement workshop.

The workshop leaders thought that the value-requirement workshop was successful in several respects. First, the participants seemed to understand the main ideas of VSD and what was expected of them in the workshop. Second, the participants were able to accomplish all steps in the workshop. Third, the workshop evoked discussions and information exchanges among the participants, and that seemed to contribute in developing a common view on the goals of the IQmulus projects. A potential drawback of the technique is that it requires a considerable amount of time. In this workshop, 13 stakeholders were identified, but in the total duration of the workshop, 4 hours, only 3 stakeholders were analyzed.

5.5. Discussion and conclusions

In this paper, we introduced our approach to integrate values-oriented techniques with a requirements engineering process. Our approach helps requirements en-

gineers and stakeholders identify values and translate them into a form that is suitable for use in the further requirements engineering process. In a case study, we demonstrated that our approach yields value stories – user stories, a concept familiar in requirements engineering, that include values.

Future work should evaluate this approach in a design process, focusing on its ease of use and usefulness. In particular, evaluations should aim to assess the extent to which the approach helps designers without experience in dealing with values identify and understand value issues, and successfully formulate requirements to address these issues. Furthermore, future work can further support the integration of values into further requirements engineering activities by developing ways to formalize values and their relationships with other entities that currently figure in requirements specification.

Another promising direction for future work is the development of tool support. Tools could provide functions to support documentation of identified values, value stories, and related concepts, as well as provide means to visualize the resulting structure. By using such tools across design projects, designers could build up a reusable knowledge base or design patterns of values, containing the range of values encountered, as well as the requirements formulated to support these values.

References

[1] B. Friedman, P. Kahn Jr, and A. Borning, *Value sensitive design and information systems,* Human-computer interaction in management information systems: Foundations **4** (2006).

[2] A. Borning, B. Friedman, J. Davis, and P. Lin, *Informing public deliberation: Value sensitive design of indicators for a large-scale urban simulation,* in *EC-SCW 2005*, edited by H. Gellersen, K. Schmidt, M. Beaudouin-Lafon, and W. Mackay (Springer Netherlands, 2005) pp. 449–468, 10.1007/1-4020-4023-7_23.

[3] A. Czeskis, I. Dermendjieva, H. Yapit, A. Borning, B. Friedman, B. Gill, and T. Kohno, *Parenting from the pocket: value tensions and technical directions for secure and private parent-teen mobile safety,* in *Proceedings of the Sixth Symposium on Usable Privacy and Security*, SOUPS '10 (ACM, New York, NY, USA, 2010) pp. 15:1–15:15.

[4] L. Nathan, P. Klasnja, and B. Friedman, *Value scenarios: a technique for envisioning systemic effects of new technologies,* in *CHI'07 extended abstracts on Human factors in computing systems* (ACM, 2007) pp. 2585–2590.

[5] J. K. Miller, B. Friedman, and G. Jancke, *Value tensions in design: the value sensitive design, development, and appropriation of a corporation's groupware system,* in *Proceedings of the 2007 international ACM conference on Supporting group work*, GROUP '07 (ACM, New York, NY, USA, 2007) pp. 281–290.

[6] B. Friedman and D. Hendry, *The envisioning cards: a toolkit for catalyzing humanistic and technical imaginations,* in *Proceedings of the SIGCHI Conference*

on Human Factors in Computing Systems, CHI '12 (ACM, New York, NY, USA, 2012) pp. 1145–1148.

[7] D. Zowghi and C. Coulin, *Requirements elicitation: A survey of techniques, approaches, and tools,* in *Engineering and managing software requirements* (Springer, 2005) pp. 19–46.

[8] B. Nuseibeh and S. Easterbrook, *Requirements engineering: a roadmap,* in *ICSE '00 Proceedings of the Conference on The Future of Software Engineering* (ACM, 2000) pp. 35–46.

[9] S. Thew and A. Sutcliffe, *Investigating the role of 'soft issues' in the re process,* 16th IEEE International Requirements Engineering, 2008. RE'08 , 63 (2008).

[10] S. H. Koch, R. Proynova, B. Paech, and T. Wetter, *How to approximate users' values while preserving privacy: experiences with using attitudes towards work tasks as proxies for personal value elicitation,* Ethics and information technology **15**, 45 (2013).

5

6

Principles for Value-Sensitive Agent-Oriented Software Engineering

As software plays an increasingly important role in people's lives, the impact it has on their values frequently becomes apparent. Many software design methods address "soft issues", but very few ad- dress values explicitly. We present six principles that design methods should meet in order to properly deal with values. One area in which adherence to stakeholder values is important, is Agent-Oriented Soft- ware Engineering (AOSE). The Tropos AOSE method, with its concept of soft-goal, comes close to meeting our principles, but does not address values explicitly. Value-Sensitive Design is a methodology that does ex- plicitly address value issues, but it offers little guidance in operationaliz- ing them. We discuss a case study in which we attempt to cap- ture values in Tropos' soft-goals after eliciting them using Value-Sensitive De- sign. Subsequently, we discuss to what extent Tropos adheres to our princi- ples. Finally, we propose the introduction of values as a first-class entity in Tropos in order to meet our aims of dealing with values.

6.1. Introduction

In 2009, the designers of the social networking website Facebook introduced a number of changes to the website. Due to these changes, users were no longer able to choose with whom they shared the list of people (their "friends") they were connected to on the website. Anyone logged in to Facebook could now see to whom any member of the website was connected. Further, Facebook decided users' profile pictures and the pages they "like"[2] were now publicly accessible information. That is, information users shared on the website regarding their interests would now be available to the Internet at large.

Facebook (partially) violated two values by introducing these changes. The changes violated users' value of autonomy by giving them diminished control over how their information is shared. Furthermore, the fact that certain personal information was now public impacted users' privacy. The violation of these values led to user outrage and criticism by organizations such as Electronic Frontier Foundation [1]. Facebook responded that it tried to uphold the value of openness shared by their target audience [2].

This conflict of values and the way it became clear exemplifies the problem we seek to address in this paper. Designers necessarily impart social and moral values in making choices in the design of information systems [3]. That is, designers' values, such as openness, are "put into" software artifacts, albeit implicitly. Once a system has been put into use, it affects its stakeholders by supporting or hindering their values to various degrees. This ultimately affects the acceptability of information systems. Often, these values and value issues only become explicit after the software has been put into use, at which point the damage has been done. Therefore, we plead for dealing with values explicitly by treating them as separate "first-class entities" throughout the design process in software design methods.

This problem holds for software engineering in general, but is especially relevant in agent-oriented software engineering (AOSE), where we design agent-based systems. These systems are autonomous, reactive, pro-active, and have social ability [4]. Moreover, they act on stakeholders' behalf, so it is important that they meet stakeholders' requirements. Values can be considered requirements in that they are stakeholder needs that systems should uphold. The issue of meeting requirements is part of one of the areas of AOSE research identified in [5]. Weyns et al. conclude that we have to extend our research into goal-oriented design, verification and validation in order for agent-oriented software engineering to be adopted in industry. In particular, we should be able to provide guarantees with respect to stakeholder requirements.

In as far as design methodologies explicitly take values into account in the design process, it is in the form of non-functional requirements [6] or similar constructs. However, designers run the risk of leaving the impact values have on design implicit by representing values as non-functional requirements. Methods such as Quality Attribute Workshops and its notion of scenarios [7] and Attribute-Driven Design [8] deal with non-functional requirements formally, and have been applied to AOSE by

---

[2]Web pages on Facebook about topics, people, places, books, etc. that people can"like".

Weyns [9]. However, AOSE methods typically neglect non-functional requirements. Values should be included explicitly as entities in themselves in order to be properly considered and to have an identifiable and justifiable effect on the design.

AOSE methodologies such as Tropos do focus on stakeholders' requirements (in the form of goals) throughout the design process, but do not explicitly take values into account. To address values in Tropos, they have to be represented as (soft)goals. Many important characteristics of values are lost in representing values as goals. Value-Sensitive Design [10] (VSD) provides a comprehensive framework for eliciting values, but provides little to make these values concrete. In this paper we propose to address this issue by combining elements of VSD with Tropos. This should form an AOSE approach that meets our aim of making the influence of values on the design explicit during all design phases.

This paper is organized as follows. In section 2 we briefly discuss the concept of values and discuss six value-related principles design methods should adhere to. We then discuss some common ways of dealing with such issues in requirements engineering. Then, in section 3 we discuss a case study to discover to which extent values can be dealt with in Tropos. In section 4, we analyze to what extent Tropos adheres to our principles and discuss important differences between values and the soft-goals we use to include values in Tropos, and propose introducing a value entity in Tropos. We draw conclusions and suggest directions for future work in section 5.

6.2. Values in Existing Software Engineering Methods

This section discusses to what extent values are already taken into account in existing software engineering approaches with an emphasis on agent-oriented software engineering methods, and in particular Tropos. Before this discussion, we present an overview of the concept of values and discuss the role of values in relation to the stakeholders and designers in the design of multi-agent systems. Finally, we provide a short introduction of the Value Sensitive Design method and discuss why VSD is not an answer in itself.

6.2.1. Values

The introduction describes a real world case of stakeholders' values (i.e., privacy, autonomy and openness) being hindered or supported by technology. Other values implicated in system design include human welfare, ownership and property, freedom from bias, and trust [10]. The general notions of norms and values are known to us all; norms and values are instilled into all of us during our childhood by our parents and social surroundings and continue to be throughout our lives.

Values are abstract (e.g., [11, 12]), motivational constructs that apply across contexts and time [11]. They convey what is good (e.g., [13, 14]) and important to us (e.g., [10, 11]). For example, privacy was something good and important for users in the Facebook case. As a result, they were outraged when their privacy was not respected. They would have reacted similarly if another website, person

or institution had failed to respect their privacy, as values hold across situations. As Hodges and Baron argue, values are convictions that some things ought to be and others not [15]. To make the concept of a value more precise it can be differentiated from similar concepts, such as laws, rules, goals, norms, standards, and so on (e.g., [12–14, 16, 17]).

Values have a special status due to their importance to their holders (violation of values is seen as deplorable or morally wrong) and the expectations they generate regarding the behavior of the holder and of others. Values create preference for behavior or action that supports them, which gives them a normative character. As Miceli and Castelfranchi point out regarding the normative character of values, "if something is good, it should be pursued" [13, p. 181]. For example, "honesty" is a value which gives rise to a norm "be honest". Moreover, if something is good, it should not only be pursued by the holder of the value; it should be pursued by others as well. However, others do not always hold the same values. This normative character of values is a ground for conflicts when people hold different values or different priorities among their values.

Our work is concerned with the design of multi-agent systems and systems that are expected to have a social impact. Considering that the systems we build can conflict with the values and norms of the stakeholders of these systems, it is especially important to explicitly recognize the role values play in design.

Returning to the Facebook example, we can say that the value openness of Facebook gave rise to a norm of the Facebook team, i.e., "everybody should share personal information", which conflicts with the value of privacy of the users. In retrospect, could we not say that the way out of this conflict lies in considering the shared value of autonomy, with an associated norm that everybody should be able to decide for herself? Based on this shared norm we can derive the more specific norm that everybody has to decide for herself whether to share information or not. It is a compromise between openness and privacy that is acceptable to both developers and users of Facebook.

This example illustrates the abstract and normative nature of values. Values can be instantiated according to the situation at hand. For example, the value of autonomy is instantiated to insisting on control over how to share personal information on Facebook. The dormant problem of two conflicting abstract values (openness and privacy) became acute at the instantiated level. This leveled approach, working with instantiations, can also be found in the work of Maio [12]. To discover possible conflicts at an early stage of system development, we advocate value elicitation at the start of the project to make people consciously aware of their values; this will reduce costs, effort, and frustration. Proynova et al. make a similar plea [18], focusing mainly on elicitation.

We recognize that, though conflicts between moral values are not dealt with as such in the approaches described here, many mainstream software engineering methods do deal with conflicts of a similar form. Certain design decisions may hinder one value while achieving another. Conflicts with this structure are dealt with in mainstream software engineering methods in the form of tradeoffs between quality requirements (see, for example, [9]).

In our opinion the process of value elicitation at the start of a design process should answer the following questions. Which people's values can be impacted by the system under design and which people's values can impact the design of the system? In our view, this question is essential for the design of system and its answer is both obvious as well as treacherous by its obviousness. The answer to the first part of the question is the stakeholders, and the answer to the second part is the stakeholders and the designers/developers. The last addition, that of the designers, is easy to overlook, as the designers might unconsciously assume that their values are shared by the stakeholders. The Facebook example is illustrative of this point. We conclude that to avoid the negative consequences of violating values and to promote the support of values as much as possible, the following principles should be satisfied by design methods.

1. The values of all stakeholders including designers/developers should be elicited in as far as relevant for the system under design.

2. Stakeholder values should be addressed during all phases of the design process.

3. Conflicts between values of the designers and those of the stakeholders need to be discussed with those who issued the order for the system.

4. To account for the relevant values, to the relevant values need to be instantiated explicitly throughout the design process.

5. Design decisions can and need to be justified and evaluated in terms of explicit (instantiations of) stakeholders' values.

6. Conflicts between values need to be made clear and addressed in cooperation with the stakeholders.

These principles are used in the next section to discuss how existing requirement engineering methods as part of design methods deal with values.

6.2.2. Requirements and Values

Requirements engineering is one of the first steps in the larger process of software development. It is the process of identifying stakeholders and their needs, and documenting these in a form that can be analyzed, communicated, and subsequently implemented [19]. Broadly speaking, there are two types of requirements: functional requirements and non-functional requirements [6]. The former are requirements that define a function of the system, or something that a system will do. The latter define not what a system will do, but how it will do it. Requirements engineering has attention for "soft issues" such as politics and people's values, although dealing with soft issues is problematic as there is little guidance on how to do so [20]. Concepts used to specify soft issues include *non-functional requirements*, *quality attributes*, *soft constraints*, and *soft-goals*.

Though there is no consensus in the requirements engineering community as to exactly what non-functional requirements are [21], broadly speaking a non-functional requirement is "a software requirement that describes not what the software will do, but how the software will do it" [6, p. 6]. Non-functional requirements are often refered to as "-ities" or "-ilities" [22]. Examples of non-functional requirements include usability, maintainability, adaptability, efficiency, and flexibility.

The concept of non-functional requirement appears to be broad enough to cover values. In fact, some values, namely security and privacy (as a feature of security), have been dealt with in an extension of the Tropos method [23]. However, not all non-functional requirements are values. Non-functional requirements such as maintainability or portability, while important, are conceptually far removed from the moral good worth pursuing that values such as autonomy, trust, and justice point to. The examples of non-functional requirements given here are closely related to the envisioned system, whereas the examples of values are more closely related to humans, culture, or society. Furthermore, as far as we know, no specific guidelines exist for dealing with moral values in design methods that use the concept of non-functional requirements.

The related concept of *quality attribute* can be defined as "[a] feature or characteristic that affects an item's quality" where quality is understood as "[t]he degree to which a system, component, or process meets specified requirements" or "[t]he degree to which a system, component, or process meets customer or user needs or expectations" [24, p. 60]. As with non-functional requirements, this term is so general that it provides no guidelines for dealing with values specifically.

Soft constraints are requirements for dealing with over-constrained problems, as well as for dealing with uncertainty, vagueness or imprecision [25]. As stated in [25]: "They can be seen as a preferential constraint whose satisfaction is not required, but preferred." Treating soft constraints as "preferred but not required" disqualifies soft constraints as the way to model values as the moral wrongness of violating a value is lost. Nonetheless, we can try dealing with values as soft constraints. Soft constraints are to be elicited during the requirements engineering process, however, if values are not specifically addressed chances are that no values will be made explicit (principle 1). Soft constraints of stakeholders are typically taken into account, and that way principle 2 can be said to hold in as far as principle 1 is upheld. Principle 3 is not treated using values. Principles 4, and 5 are treated accepting that values are part of the whole set of soft constraints. Principle 6 is not dealt with as such.

Soft-goals, as used in e.g., Tropos [26], are requirements that are not clearly defined and do not have clear criteria for satisfaction, drawing on the notion of satisficing instead [27]. They are a form of non-functional requirements that refer explicitly to goals, an important concept in agent technology.

As we are particularly interested in agent-oriented software engineering [28] we focus on Tropos and its soft-goals. Treating values as soft-goals, we can summarize that principles 1, 2, 4, and 5 are treated to some extent in Tropos, but principles 3 and 6 are in no way part of the Tropos method. With respect to principle 1, indirect stakeholders are not taken into account, although the method could be

easily adapted to cover this. Principle 2 is covered in the sense that soft-goals can play a role during all phases of the design. Principle 5 is covered in the sense that decisions are related to soft-goals, but not in as far as one soft-goal is weighed more heavily than another to make a choice.

Section 3 describes our effort to see how far we can get with modeling values as soft-goals in Tropos and will explain our conclusions regarding the principles.

Before focusing on Tropos and the possibilities soft-goals offer to include values in the design, we would like to mention one more approach that might be useful with respect to values.

6.2.3. Value-Sensitive Design

VSD "is a theoretically grounded approach to the design of technology that account for human values in a principled and comprehensive manner throughout the design process" [29]. In VSD, emphasis is given to supporting moral values or values with ethical import, such as human welfare, ownership of property, privacy, and freedom from bias [10].

VSD provides an iterative and integrative three-part methodology consisting of conceptual, empirical, and technical investigations. Conceptual investigations focus on discovering affected stakeholders, their values, and analyzing these values and tensions between them [30]. The first step is to perform a stakeholder analysis to identify direct and indirect stakeholders, which are the people who interact directly with the technology, and those who are impacted by the technology without interacting with it, respectively.

For each group of stakeholders, potential harms and benefits are identified. The list of harms and benefits can be used to map harms and benefits onto associated values, especially human values with ethical import.

Once these key values have been identified, a conceptual investigation of the values is conducted supported by (philosophical) literature, resulting in clear definitions of those values. Potential value conflicts, which can constrain the design space, are examined. Stakeholders are involved if conflicting values hinder one another in the design, such as accountability versus privacy.

Conceptual investigations need to be informed by empirical investigations of the technology's context. VSD does not prescribe a specific method for this stage, stating that "the entire range of quantitative and qualitative methods used in social science research is potentially applicable" [10]. Friedman and colleagues do suggest that semi-structured interviews of stakeholders can be a useful method to understand stakeholders' judgments about a context of use, an existing technology, or a proposed design.

Technical investigations focus on the properties and mechanisms of existing technologies that support or hinder human values. Alternatively, technical investigation can consist of designing a system to support identified human values. Though technical investigations of the first form and empirical investigations seem similar, technical investigations focus on the technology itself, and not on the individuals affected by it, as empirical investigation does. During this stage, it can be helpful to make explicit how design trade-offs map onto value conflicts and affect

different groups of stakeholders.

It could be argued that, individually, the steps taken in VSD are common sense. Common sense as they may be, these steps are rarely taken together in a structured manner. As a result values are often neglected in design and addressed after the fact, as cases of privacy issues with social networking websites, bias in search engines, and intellectual property issues with file-sharing software illustrate. VSD offers a structured approach to addressing values.

The strengths of VSD lie in its focus on direct and indirect stakeholders, how they are or will be affected by the technology, and what values are implicated. The focus on a broad range of stakeholders, along with the identification of potential value conflicts and the aim to deal with values throughout design, suggest that VSD adheres to our six principles. However, VSD would benefit from means to not just elicit values, but actually incorporate them in design and eventually implement them.

6.3. Case Study: Values in Tropos

To discover to which extent values can be dealt with in Tropos in adherence to the six principles of Section 2.1 we performed a case study. The chosen case study is that of designing a conference management system with an emphasis on the values involved. We picked this case study as it was used in [28] to illustrate the use of three agent-oriented software engineering methods, including Tropos, and was based on an earlier case study presented in [31]. Furthermore, conference management systems are at the core of the peer-reviewing established by researchers to protect the quality of research. The decisions made during peer-reviewing have a high impact on researchers. Therefore, the design of such a system must be done in such a way that the norms and values of the stakeholders are respected as much as possible.[3]

The rest of this section is organized as follows. We first identify Tropos, we then describe the general purpose of conference management systems and identify the stakeholders, after which we inject the process of value elicitation for use later on. We then proceed with the remaining value-related steps in the Tropos method with an emphasis on how values are addressed in these steps.

The Tropos software development methodology supports the agent-oriented paradigm and the associated concepts of actors, plans and goals throughout the software development process [26], [28], [32]. Its main value-related steps are stakeholder identification, goal identification, and goal decomposition.

The general purpose of a conference management system depends on the stakeholders involved and vice versa. Tropos identifies stakeholders early in the

---

[3]Note that the design of a conference management system in terms of the roles involved is primarily determined by the organization structure of the conference. In this case we chose for a conference management system that adheres to that of smaller conferences or workshops and ignored the more recent use of a Senior Programme Committee as is used in the AAMAS conference. It would be good practice to design the organization structure of the conference before designing the conference management system. However, for our purpose of showing how to deal with values, it is enough to start with some conference organization structure.

design process, in the Early Requirements phase. The main stakeholders involved are a paper authors, paper reviewers, program chairs, and publisher of the proceedings [31]. To this we add the general public / government and the researcher as indirect stakeholders. We assume that the fundamental choice for blind peer reviewing has already been made in the organization of the conference. The general purpose is to support paper submission, bidding for papers for review, distribution of papers to reviewers, collection of reports, supporting program committee meetings, communication of results, and submission of camera ready versions of papers. All these aspects are subservient to the underlying concern of publishing high quality research only and blocking substandard research reports. The general purpose and the underlying concern already implicitly refer to a number of values.

Value elicitation was performed with each stakeholder group and ourselves as system designers. We used semi-structured interviews as suggested in the VSD method of [10]. In the interviews we explained the intention of designing a conference management system and described the basic activities it would support. We asked stakeholders to identify potential harms and benefits of such a system, and together with them identified the values underlying these harms and benefits. It is important to note that most interviewees had experience with existing systems and due to that it is likely they were reflecting on the systems they were familiar with. Also, most interviewees had experience with multiple stakeholder roles, making it difficult to rule out that they projected values they hold in one role to another role.

The authors mention anonymity of reviewers and conflicts of interests as potential harms and anonymity of authors as a benefit. They stated that anonymity removes context, which makes it difficult to assess reviewers' expertise and damages the quality of the discussion. Also, it allows reviewers to "ride their hobby horse", posing a threat to their objectivity. On the other hand anonymity of authors removes hierarchical considerations, leading to judgments based on quality and not on academic position. This is a potential benefit. The authors warned for conflict of interests arising from users occupying multiple roles within the same system. This could lead to reviewers who are also authors seeing the ranking of their own paper or reviewers reviewing papers of friends. The authors concluded that the harms are based on their values of transparency, fairness, and accountability, while the benefits are based on their values of fairness and privacy and would improve the quality of publications.

Reviewers mentioned anonymity of reviewers as a benefit. It also allows reviewers to be as critical as (they feel) they need to be. Together, we concluded that the underlying values are privacy and quality of publications. PC chairs considered reuse of the system across conferences to be a potential benefit, which contributes to the trustworthiness of the system. Trust is the underlying value here. A potential harm that one PC Chair identified was the potential for bias in seeing authors' names. This could lead to bias based on gender and ethnicity.

Publishers benefit from the peer review process the system supports. By publishing high-quality research and barring substandard research, the reputation of the publication and that of the publisher potentially increase, as do sales. This supports publishers' values of quality, profit and trust.

Researchers in general consider it a potential harm that poor quality research is disseminated. Poor quality research is damaging to the reputation of the research community with the general public and with government. Also, if researchers' own work is disseminated and of poor quality, it is damaging to their reputation with peers. Both senses of reputation, and the related value of scientific integrity, are values held by researchers.

The general public and government see the publication of high quality research and the barring of sub-standard level publications as potential benefits. These ultimately support the value of knowledge.

As system designers in this case, we discovered that we were influenced by our identification with the roles of author, reviewer, and PC chair, and as such shared many of the values of those stakeholders groups.

All stakeholders identified use of the conference management system for multiple conferences as a benefit. Reuse enhances the trustworthiness of the system and the process it supports. Also, the record of interactions with the system supports transparency and accountability.

In summary, we can see a range of values at stake here, among which potential or real conflicts exist, for example between transparency and privacy. This example conflict leads to opposing views on whether the system should provide anonymity. A compromise between such values must be found, that is, a feature that supports both or at least hinders neither.

Stakeholders' goals are identified next, and for every goal the developer decides whether the actor itself can achieve it or whether it needs to be delegated to another actor. Goals represent strategic interests of actors. A distinction can be made between (hard) goals and soft-goals. Hard goals have clear criteria for satisfaction. Soft-goals do not have clear criteria for satisfaction, drawing on the notion of satisficing instead [27].

The only option that Tropos has for representing the values identified in the previous stage are soft-goals. Due to space limitations, we will only discuss how the potential harm/benefit of anonymity, the potential harm of conflicts of interest and the underlying values at stake could be addressed in Tropos. Tropos actors are written in italicized bold. Goals and soft-goals are written in bold.

Authors saw the anonymity of reviewers as a potential harm as it prevents them from assessing the expertise of the reviewer. So, we could say the *Author* has a goal, **know reviewer identity**, which contributes positively to the values of **transparency** and **accountability**. The **know author identity** goal is why-linked to a goal dependency between the *Author* and the *PC Chair*, **disclose reviewer identity**. We will discuss how this conflicts with *Reviewers*' goals shortly.

Authors saw their own anonymity as a potential benefit. So, we introduce the goal **anonymity from reviewers**. This goal contributes positively to the *Author*'s values of **privacy** and **fairness**, which we represent as soft-goals. The goal is why-linked to the goal dependency **protect author anonymity** between the *Author* and the *PC Chair*.

Authors also saw conflicts of interest as a potential harm. So, the *Author* actor depends on the *PC Chair* to **avoid conflicts of interest**. This goal contributes
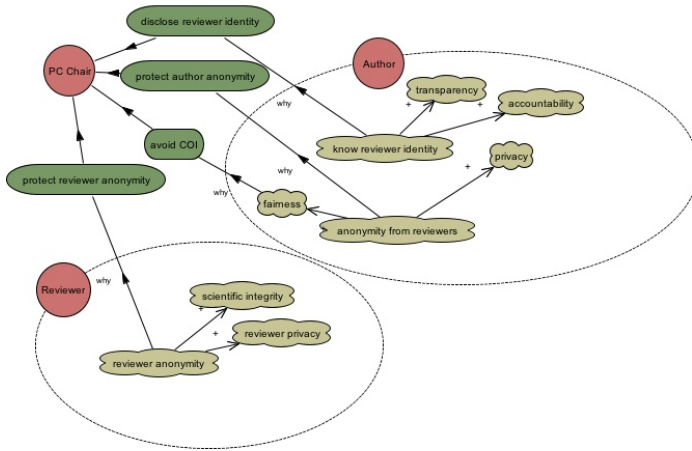
Figure 6.1: Early requirements model of conference management system in Tropos with values as represented as softgoals

positively to the value of **fairness**, represented as a soft-goal. However, since **avoid conflicts of interest** is a goal dependency and hence becomes the *PC Chair*'s goal, the only option we have to link it to the *Author*'s value of **fairness** in Tropos is the why-link.

Reviewers saw anonymity as a potential benefit. Therefore, we say that the *Reviewer* actor has a goal dependency, **protect reviewer anonymity**, on the *PC Chair* actor. This contributes positively to the *Reviewer*'s values of **scientific integrity** and **privacy**, represented as soft-goals. Since the **protect reviewer anonymity** is a goal dependency, the only option we have to indicate the link between it and the values it contributes to is the why-link. However, the why-link is also a type of dependency, and only one link can be constructed for a dependency. So, we have to define an intermediate goal, **reviewer anonymity**, which contributes positively to **scientific integrity** and **privacy** and is why-linked to **protect reviewer anonymity**.

Reviewers' goal **protect reviewer anonymity** obviously conflicts with authors' goal to **know reviewer identity**. Reviewers' value of **privacy** conflicts with authors' value of **transparency** here.

We attempted to model values in Tropos as soft-goals in order to meet the aims expressed in our six principles. However, there are a number of issues with this that we will discuss in the next section.

6.4. Discussion

6.4.1. Six Principles

We will now discuss the results of the case study described in section 3 in light of the six principles described in section 2.

The first principle states that the values of all stakeholders and designers or developers should be elicited as far as relevant to the system under design. While stakeholders are considered in Tropos, the group of stakeholders considered is limited to actors that will eventually use the system in some way. Indirect stakeholders, such as the general public in the case study above, are not considered, though they may be affected by the (output of the) system. Also, designers and developers are not considered in Tropos.

The second principle states that stakeholder values should be addressed throughout the design process. As the case study demonstrates, if we represent values as soft-goals in Tropos, then they can be said to be addressed throughout the design process. However, as we discuss below, values are not (soft) goals.

The third principle states that conflicts between the values of the designers and those of the stakeholders need to be discussed with those who issued the order for the system. Since Tropos does not consider the designers as such, conflicts between their values and those of the stakeholders do not become apparent.

The fourth principle states that values have to be instantiated explicitly throughout the design process. If we represent values as soft-goals, we can say that values are instantiated throughout the design process through the process of goal decomposition. However, there are problems with treating values as goals, which we discuss below.

The fifth principle is that design decisions need to be justified and evaluated in terms of explicit instantiations of stakeholders' values. We can say that goals and decompositions of goals into lower-level goals are design decisions. By drawing contribution links between these goals and soft-goals representing values, we can in a sense evaluate and justify these design decisions by seeing which design option (alternative subgoal) contributes best to the soft-goal (value) in question. It should be noted that the extent to which contribution can be expressed is limited. The metrics + and ++ indicate partial and sufficient positive contribution, respectively, and the metrics − and −− indicate partial and sufficient negative contribution, respectively [26].

The sixth principle states that conflicts between values need to be made clear and addressed in cooperation with the stakeholders. In Tropos, the only links between (soft) goals are varieties of decomposition links, namely AND or OR decompositions, means-end links, or contribution links. Also, only one link can exist between these (soft) goals. That is, we cannot have a goal 1 contribute to a goal 2, and have that goal 2 contribute to goal 1. Therefore, we cannot express conflict between (equally abstract) values as such, for example openness and privacy. We could define a higher level soft-goal (value) and say that one lower-level soft-goal contributes positively to it, while another contributes negatively. These soft-goals would then be in conflict, in terms of how they contribute to the higher-level soft-goal, but this is not an option for intrinsic values (or ends) in conflict.

In summary, we can say that the first, fifth and sixth are satisfied to some extent; the second and fourth are satisfied if we consider values to be goals; the third cannot be said to be satisfied. However, this is the very reason why Tropos does not adhere to our principles. To adhere to the principles we would have to

represent values as soft-goals, but values should not be treated as soft-goals.

6.4.2. Differences Between Values and Goals

Values are not the same thing as goals. Miceli and Castelfranchi provide a useful distinction between these concepts. "Values are not goals, they are assumptions (more precisely, evaluations). A value is a judgment, though very general and vague. It says of something that it is good or bad. A goal is a regulatory state in someone's mind" [13, p.179]. They illustrate a further important feature of values in discussing the difference between values and norms: "Values in fact offer grounds for, or give rise to norms. Hence the 'normative' facet of values: If something is good, it should be pursued" [13, p. 181]. If we represent values as soft-goals, the evaluative aspect ("X is good") and the normative aspect ("X should be pursued") are lost. Represented as a soft-goal, a value becomes something that can be satisfied (i.e., sufficiently satisfied). Not achieving a goal is not morally wrong as such. Violating a value is seen as morally wrong. This distinction is important. Not taking these aspects into account could lead to problems once the design has been implemented and put into practice, as we saw in the example of Facebook.

6.4.3. Dealing with Values in Tropos

Considering the issues with representing values as soft-goals, we propose some additions to the Tropos approach. First of all, in line with our first principle, we propose that the notion of stakeholder in Tropos be extended beyond those groups that delegate their goals to a system to all who will be affected by the system (i.e., direct and indirect stakeholders) and those who shape the system. These groups of stakeholders need to be approached as a source of requirements (values and otherwise) early in the requirements engineering process.

Second, since values should not be represented as goals, we propose the addition of a first-class value entity to Tropos. Since values are held by stakeholders, the value entity needs to be connected to the stakeholders that hold it. As we discussed above, values are general and abstract evaluations. They are conceptions of what is good and are important to their holder. We need to be able to indicate the goodness and importance of each value to its holder in some way, so we can prioritize values and assess the importance of addressing each one. Further, since what is good should be pursued, values can give rise to goals and norms. Hence, we need to be able to represent links between values and the norms and goals they generate. Norms should also be represented, but this is beyond the scope of this paper. Values eventually need to be implemented in some way. Antunes and Coelho's Belief, Values, Goals (BVG) architecture uses values as central motivational mechanisms in their agents' minds [33]. We see this as even more of a motivation to address values early on in design. Also, designers could make use of such an architecture to implement the values elicited and represented during the requirements phase.

Third, values and their instantiations can conflict. The conflict between Facebook's value of openness and users' value of privacy is a case in point. We need to be able to identify such conflicts in order to deal with them early on. To this

end, we propose the addition of a conflict relationship between entities, specifically values, in Tropos.

Including indirect stakeholders as a source of (value) requirements, treating values as separate entities in models, explicitly representing conflicts between values, and dealing with values throughout design, as implementing our proposals will allow us to do, will provide us with an approach that adheres to the six principles described in section 2.1.

6.5. Conclusions

In summary, software impacts human values. In light of this fact and the special status values have, we proposed six principles designers should adhere to. Some requirements and software engineering concepts seem similar to values. However, there are some important differences between values and these concepts. VSD is a methodology that aims to account for (moral) values in design. VSD is a useful methodology for eliciting and defining stakeholders' values. However, VSD as-is does not provide a means for implementing such values. This makes it difficult to assess the extent to which values are incorporated in actual designs.

In our case study in we attempted to capture values in Tropos soft-goals and showed that Tropos as-is cannot fully handle our six principles. We argued that Tropos' soft-goals are fundamentally different from human values as described here. Representing values as soft-goals does not make values sufficiently explicit.

To address these problems, we propose complementing Tropos with a separate, first-class entity to capture values. This entity will allow the designer to explicitly represent values throughout the design process, and to make values concrete enough to operationalize them and to expose and address conflicts between them.

Future work should address the issue of representing values. Also, future work should deal with representing and addressing value conflicts, as these are an important source of many of the issues with values in design. To this end, a formal framework of values is needed. Further, the issue of dealing with different stakeholders' views on specific values should be addressed.

References

[1] K. Bankston, *Facebook's new privacy changes: The good, the bad, and the ugly,* (2009).

[2] M. Kirkpatrick, *Facebook's zuckerberg says the age of privacy is over,* (2010).

[3] B. Friedman, *Human Values and the Design of Computer Technology* (Cambridge University Press and CSLI, New York, NY and Stanford, CA, 1997).

[4] M. Wooldridge and P. Ciancarini, *Agent-oriented software engineering: the state of the art,* in *AgentOriented Software Engineering III* (Springer, 2001) pp. 55–82.

[5] D. Weyns, H. Parunak, and O. Shehory, *The future of software engineering*

and multi-agent systems, International Journal of Agent-Oriented Software
Engineering **3** (2009).

[6] L. Chung, B. Nixon, E. Yu, and J. Mylopoulos, *Non-functional requirements in software engineering,* (2000).

[7] M. Barbacci, R. Ellison, A. Lattanze, J. Stafford, C. WeinStock, and W. Wood, *Quality Attribute Workshops (QAW), (CMU/SEI-2003-TR-016)*, Tech. Rep. (Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, 2003).

[8] R. Wojcik, *Attribute-Driven Design (ADD), Version 2.0 CMU/SEI-2006-TR-023*, Tech. Rep. (Software Engineering Institute, Carnegie-Mellon University, Pittsburgh, PA, 2006).

[9] D. Weyns, *Architecture-Based Design of Multi-Agent Systems* (Springer-Verlag New York Inc, 2010).

[10] B. Friedman, P. Kahn Jr, and A. Borning, *Value sensitive design and information systems,* Human-computer interaction in management information systems: Foundations **4** (2006).

[11] A. Bardi and S. Schwartz, *Values and behavior: Strength and structure of relations,* Personality and Social Psychology Bulletin **29**, 1207 (2003).

[12] G. R. Maio, *Mental representations of social values,* Advances in Experimental Social Psychology, Vol 42 **42**, 1 (2010).

[13] M. Miceli and C. Castelfranchi, *A cognitive approach to values,* Journal for the Theory of Social Behaviour **19**, 169 (1989).

[14] M. Schroeder, *Value theory,* in *The Stanford Encyclopedia of Philosophy*, edited by E. N. Zalta (2008) fall 2008 ed.

[15] B. H. Hodges and R. M. Baron, *Values as constraints on affordances - perceiving and acting properly,* Journal for the Theory of Social Behaviour **22**, 263 (1992).

[16] M. Rokeach, *Beliefs, attitudes and values: A theory of organization and change* (1968).

[17] J. Spates, *The sociology of values,* Annual Review of Sociology **9**, 27 (1983).

[18] R. Proynova, B. Paech, A. Wicht, and T. Wetter, *Use of personal values in requirements engineering–a research preview,* Requirements Engineering: Foundation for Software Quality , 17 (2010).

[19] B. Nuseibeh and S. Easterbrook, *Requirements engineering: a roadmap,* in *ICSE '00 Proceedings of the Conference on The Future of Software Engineering* (ACM, 2000) pp. 35–46.

6

[20] S. Thew and A. Sutcliffe, *Investigating the role of 'soft issues' in the re process,* 16th IEEE International Requirements Engineering, 2008. RE'08 , 63 (2008).

[21] M. Glinz, *On non-functional requirements,* in *Requirements Engineering Conference, 2007. RE'07. 15th IEEE International* (IEEE, 2007) pp. 21–26.

[22] L. Chung and J. do Prado Leite, *On non-functional requirements in software engineering,* Conceptual Modeling: Foundations and Applications , 363 (2009).

[23] H. Mouratidis and P. Giorgini, *Secure tropos: A security-oriented extension of the tropos methodology,* International Journal of Software Engineering and Knowledge Engineering **17**, 285 (2007).

[24] *Ieee standard glossary of software engineering terminology,* IEEE Std 610.12-1990 (1990).

[25] R. Bartak, *Modelling soft constraints: a survey,* Neural Network World **12**, 421 (2002).

[26] P. Bresciani, A. Perini, P. Giorgini, F. Giunchiglia, and J. Mylopoulos, *Tropos: An agent-oriented software development methodology,* Autonomous Agents and Multi-Agent Systems **8**, 203 (2004).

[27] E. S. K. Yu, *Towards modelling and reasoning support for early-phase requirements engineering,* Re '97 - Proceedings of the Third IEEE International Symposium on Requirements Engineering , 226 (1997).

[28] S. DeLoach, L. Padgham, A. Perini, A. Susi, and J. Thangarajah, *Using three aose toolkits to develop a sample design,* International Journal of Agent-Oriented Software Engineering **3**, 416 (2009).

[29] B. Friedman, D. Howe, and E. Felten, *Informed consent in the mozilla browser: implementing value-sensitive design,* in *System Sciences, 2002. HICSS. Proceedings of the 35th Annual Hawaii International Conference on* (2002) p. 10 pp.

[30] J. K. Miller, B. Friedman, and G. Jancke, *Value tensions in design: the value sensitive design, development, and appropriation of a corporation's groupware system,* in *Proceedings of the 2007 international ACM conference on Supporting group work*, GROUP '07 (ACM, New York, NY, USA, 2007) pp. 281–290.

[31] P. Ciancarini, O. Nierstrasz, and R. Tolksdorf, *A case study in coordination: Conference management on the internet,* (1998).

[32] F. Giunchiglia, J. Mylopoulos, and A. Perini, *The tropos software development methodology: processes, models and diagrams,* Lecture Notes in Computer Science , 162 (2003).

[33] L. Antunes and H. Coelho, *Redesigning the agents' decision machinery,* Affective interactions , 121 (2000).

6

7

Stances and Acceptance in Personal Informatics

In this paper we investigate the role of user personality and values in relation to factors influencing the use of apps for healthy food consumption. This research is motivated by the observation that these apps employ diverse strategies, which are more or less prescriptive, to support healthy eating behaviors. The choice of strategy reflects the designer's stance on how behavior change can be achieved. Judgments on values such as autonomy underlie these stances and we argue, that an alignment of user values and personality and designer stances will lead to more successful app use and, ultimately, sustainable health behaviors. The presented pilot study provides the first step of a larger research on designing technology-based behavior change accounting for user values and personality.

7.1. Introduction

Healthcare systems are shifting from treatment-centered approaches, where patients follow doctors' orders, to approaches focused on prevention and healthy living, where patients ought to take responsibility for their own health [1]. People's eating behavior is an especially important area to address. Nutrition is increasingly recognized a major determinant of chronic disease, such as cancer, cardiovascular disease or diabetes, and dietary changes have strong effects on health [2]. Personal informatics technologies are a promising way of supporting this shift in responsibility, through self-tracking and analysis of consumption behavior. Persuasive technologies could be similarly helpful in their ability to stimulate behavioral change (towards healthy consumption) [3]. Many of these technologies are implemented as mobile apps to ensure data collection continuously and in-situ.

A major challenge in designing successful mobile health intervention lies in supporting the entire process of achieving a healthier lifestyle including: creating awareness, changing unhealthy behavior, maintaining behavioral changes, and consolidating a healthy lifestyle (e.g., [4]). For this users need to be willing to use apps over longer periods of time and not only change behaviors, but also change attitudes.

Existing apps use a variety of strategies to meet the same aim of promoting healthy eating. Strategies range from providing food product information to fully tracking consumption and recommending alternative food products to choose. Some of the differences between strategies stem from considerations such as persuasive effectiveness and usability [5]. In many cases, though, different strategies reflect the designers' positions on issues such as paternalism, awareness, and education. For example, apps that take a prescriptive approach to promote healthy living (see [6] for some examples) reflect a position in support of (soft) paternalism, whereas an approach that does not prescribe reflects a less supportive (or even opposing) position. In the following we refer to these positions as designers' stances. Different judgments on values, e.g. self-direction, conformity or achievement, underlie these stances (see [7] for a similar argument). Designers' stances and users' values, however, are often left implicit in technology design (e.g., [8]). We argue that making them more transparent and aligning stances' with users' values and personality may affect how well users integrate apps into their lives and continue using them over longer periods.

In this paper we first built this argument in more detail, and then report on an exploratory pilot study that we are currently conducting to investigate how personality and user values influence different use-factors that ultimately influence the intention to use a certain app. In this study we compare three different apps ranging from prescriptive to completely open. First results of this ongoing work are presented.

7.2. Related Work

7.2.1. Personal Informatics

The rather new field of personal informatics aims to design information technology systems that help users gather and reflect on personal information, for the purpose

	C-Change	B-Change	A-Change
F-Outcome	Forming an act of complying (F/C)	Forming a behavior (F/B)	Forming an attitude (F/A)
A-Outcome	Altering an act of complying (A/C)	Altering a behavior (A/B)	Altering an attitude (A/A)
R-Outcome	Reinforcing an act of complying (R/C)	Reinforcing a behavior (R/B)	Reinforcing an attitude (R/A)

Figure 7.1: O/C-Matrix by [10]

of gaining self-knowledge [9]. Personal informatics systems let people understand and explore their own behavior, which can help encourage more healthy behaviors.

Closely linked to the Quantified Self[1] movement, personal informatics tools, i.e. mostly smartphone apps and websites, provide means to track large amounts of data about one's behaviors and to review the data. Personal informatics does not focus only on tracking health-related behaviors such as exercise or food intake, but on all aspects of life, including, e.g. tracking productivity, media consumption or web activity.

7.2.2. Persuasive Technology

While personal informatics has a strong focus on self-reflection by the user in order to understand his or her behaviors, persuasive technologies are designed with the goal to change the user's behavior using persuasion and social influence techniques [3]. Persuasive technologies often build on theories and methods developed in psychology and related fields, some with specific application domains such as the Health Belief Model and the Transtheoretical Model in the health domain.

In his recent work Oinas-Kukkonen [10] emphasizes that since information technology is never neutral "software designers ... should be well aware of the various ways and approaches how people may be, are being, and will be influenced through IT design" and presents the O/C-matrix (Figure 7.1), a framework for design and research on persuasive systems, to this end. In this matrix the three outcomes for behavior change interventions, i.e. forming, altering, and reinforcing are combined with three types of changes, namely complying, changing behavior, and changing attitudes, called C-, B-, and A-Change respectively. Attitude changes are the most difficult to achieve, but are also deemed necessary for sustained behavior changes.

7.2.3. Behavior change strategies

Persuasive systems are only one end of the design spectrum of technology-based health behavior change. [6] suggests another way that is less prescriptive but more open-ended and based on social awareness rather than persuasion. In their analysis of existing strategies to achieve change they identify four mechanisms: Prompting, Feedback and Reward, Tracking, Logging, and Diaries and Social Influence, ranging from most to least prescriptive.

Prompting involves sending messages, reminders, or feedback to the user and thereby prompting him or her to comply with a healthier behavior or check his or her health state. Feedback and Reward is a strategy that provides the user with positive or negative feedback in response to health-related choices in the form of

---

[1]http://quantifiedself.com

affirmation, condemnation, reward or punishment. Tracking, Logging and Diaries are mechanisms suggested to support a user in understanding his or her behavior, as in personal informatics systems, and thereby helping to improve health- related behaviors. Social Influence is a strategy based on social cognitive theory, which says that people model their behavior after that of the people around them. This strategy can be employed in technology-based tools by providing the users with tools to communicate and share with a peer group that aims at the same behavior change.

7.2.4. Values and Personality

When aiming for a sustainable behavioral or attitude change over time, we should look at factors that determine our behaviors and attitudes in different situations. There are a number of psychological constructs that do so, of which values and personality traits are the ones that are relatively stable over contexts and time. Values are "desirable, transsituational goals that vary in their importance as guiding principles in people's lives" [11]. In his theory Schwartz defines ten universal values that can be measured by the Portrait Value Questionnaire [11]. While values are enduring goals, personality traits are enduring dispositions. The latter describe what people are like rather than what they intent to do. However, the two mutually influence each other as confirmed by [12]. To name just a few correlations, extraversion correlated positively with achievement, stimulation and hedonism, openness to experience with stimulation and self-direction and conscientiousness positively with achievement and conformity. The authors also found "some evidence that traits have stronger influence on behavior over which individuals have little cognitive control, values on behavior under more voluntary control."

7.3. The Right App for the User

Following from the above-mentioned related work, we hypothesize that matching the underlying stances of apps as materialized by different strategies with user values and personality could have the potential to help users integrate the apps better with their lifestyle and thereby, allow for more long- term engagement and subsequently a sustained behavior and attitude change (A and B-Changes).

Currently, human computer interaction research in the area of technology-based health behaviour change focuses mainly on C and B-changes, and very little on A-changes [10], although the later are expected to lead to more long-term changes. In addition, the success of these apps is often not measured in long-term studies and their effect on lifestyles is not evaluated. One of the few long- term studies [15] found, that 27 % of the participants (N=76) found it hard to make and maintain healthy consumption changes in relation to the family, while others were generally struggling with discipline (17%). This shows, that even if people are willing to make changes, they have to be integrated with their lifestyles and social contexts in order to be effective. Situations such as parties with friends, may greatly influence people's eating behaviours. The authors concluded that "for long-term sustainability of health behaviors, it is important to link to intrinsic motivations like feeling better

or the joy or positive self-perceptions of taking care of one's health"[4].

Differences in intrinsic motivations or more general values and personality traits raise the question whether a one-fits-all behavior change app can be designed at all. Someone who is greatly influenced by their surrounding social network, or values conformity in general maybe inclined to simply follow what others do and eat. Someone valuing achievements might care more about earning badges and receiving points while following strict rules. Others, who value their self-direction may simply want to explore their food consumption without being told what to do, or what others do. These needs connect to different behavior change strategies (e.g. rewards, social awareness) named above.

From value sensitive research [8] we know that technologies can either hinder or support these user values. Therefore, we argue that an app that supports a user's values with respect to consumption behavior the better a user would be able to use such an app over time. To our knowledge, research on behavior change apps has not investigated this in detail. Also, it is often left implicit in the design of technologies, which values were explicitly supported by designers and which user values exist [8]. However, understanding this relationship in-depth would allow designers to create more effective behaviour change apps by explicitly and transparently supporting certain values that match with particular user values, and allow users to pick the app that has the highest potential to lead to a sustained change. To set out on this endeavour we started investigating the role of values and personality on other use factors in healthy consumption apps.

7.4. Pilot Study

We are conducting a pilot study to investigate the relationship of personality, values and several factors related to the use of apps that employ strategies ranging from little to very prescriptive.

7.4.1. Method

We ran an online experiment with a within-subject set-up. Each participant was exposed to interactive storyboards of three mobile apps for healthy consumption that he or she could explore. With online questionnaires we measured the personality traits and values of each participant as well as several factors related to each app, such as intention to use, usefulness, effort, feeling etc.

7.4.2. Participants

We recruited people in different ways, i.e. through social media platforms and members of the QS-movement. 28 people took part in our experiment, 14 women and 14 men, with ages ranging from 18 to 64. 75 % are interested in eating healthy, 32 % in losing weight and 18 % use smartphone apps to track their eating.

7.4.3. Materials

We selected three apps for tracking food consumption to be shown after examining a large number of mobile apps and assessing whether they include features that could

be considered a means to the Prompting, Feedback and Reward, Tracking Logging and Diaries, or Social Influence strategies. Our aim was to identify one app that is strongly prescriptive, and one that is at the opposite end of the prescriptiveness spectrum (i.e., effectively open-ended). Last, we picked one that in the middle ground. The selected apps were Noom, Evernote and Foodzy respectively.

Noom Weight Loss Coach[2] uses Prompting by providing reminders for exercise, meal-logging reminders, and daily tasks, Feedback and Reward by comparing logged meals' calorie distribution to the ideal meal for losing weight and giving a meal score (0 to 5 stars) for the day, and Rewards in form of a Noom score. The Social Influence strategy is optional. By contrast, Evernote Food[3] leaves open what healthy eating means. It allows users to document meals through photos (with caption), tags, location, meal notes, time, and selecting type of cuisine. It never tells explicitly what to do (i.e. no Prompting); all of the fields of meal entries are optional, and the app does not send reminders of any kind. It also does not use the Feedback and Rewards strategy. It does employ Tracking, Logging and Diaries and Social Influence is employed by optional sharing of meals. In terms of prescriptiveness, the Foodzy[4] application lies between Noom and Evernote Food. Users interact with the application when they please, and do not receive notifications or reminders, but occasional tips. Feedback and Reward is used in the form of visualizing the data that the user logs and with badges. Social Influence is optional by sharing via social networks.

We took screenshots of each of the three selected apps and arranged them in an interactive storyboard to represent their main functionality, i.e. enter a meal, see daily/weekly overview etc. to demonstrate the app in use.

7.4.4. Procedure

Participants we asked to fill in a demographic questionnaire (age, gender, education) followed by the TIPI [13] to measure their personality and the PVQ [11] to measure their value priorities. Then the three apps were presented as described above in randomized order. After a participants inspected all the functions of an app, they were asked to fill in the PTAM questionnaire [14]. It measures intention to use based on the factors usefulness, effort, subjective norm, support, feeling, ability and trust.

7.4.5. Data Analysis and Results

We have carried out an analysis to look for trends in the data. For the PVQ and TIPI, all constructs obtained reasonable Cronbach's alpha reliability scores except agreeableness (with a negative value violating reliability assumptions). PTAM constructs were all measured reliably (alpha from .534 to .932). A correlation analysis for values and personality found significant correlations between hedonism and conscientiousness (-.508, p=.006) and openness to experience and self-direction (.574, p=.001) as well as openness to experience and conformity (-.500, p=.007) con-

---

[2]http://www.noom.com/products/
[3]http://evernote.com/food/
[4]https://foodzy.com/about

firming parts of the original theory [11]. Before analyzing the correlations of values and personality to PTAM constructs we tested that the apps were rated differently. A repeated measures ANOVA with a Greenhouse-Geisser correction determined that mean intention to use differed significantly between apps (F(1.973, 53.269) = 8.971, p < 0.000). Over the whole population the intention to use Evernote was lowest (M=2.679, STD=1.176) while Foodzy (M=3.679, STD=1.504) and Noom (M=3.795, STD=1.612) were similar. Differences between Evernote and Foodzy (p=.002) as well as Evernote and Noom (p=.001) were significant while Foodzy and Noom ratings were not significantly different. When repeating this analysis for a) people with high openness (comprising stimulation and self-direction) and low conservation (comprising conformity and security) and b) people with low openness and high conservation, we found that in group a) the mean intention to use differed significantly between apps (F(1.731, 29.430) = 6.327, p < 0.008), in particular between Evernote (M=2.84, STD=1.18) and Noom (M=3.93, STD=1.42). In the second group of people (only 5 participants) Foodzy was scored highest (M=4, STD=1.99), but differences to other apps were not significant. For Foodzy and Noom only the PTAM factors were significant predictors of intention to use, for Evernote only usefulness (.576, p=.001), emotional stability (.491, p=0.008) and hedonism (-,401, p.034) were correlated with the intention to use. Most significantly power was correlated negatively with trust for all three apps (all above -.5).

7.4.6. Discussion and Future Work

The results show that apps were rated significantly different. However, so far few correlations between value and personality factors and the intention to use an app could be found. Most prominent is that the intention to use Evernote, the least prescriptive app, was rated low all in all groups of people. This may be based on the low perceived usefulness. The fact that hedonism was negatively correlated with Evernote hints at that participants did not perceive using this app as fun. Interestingly, people valuing power seemed to trust the apps less. Although these preliminary results cannot be used for drawing general conclusions, they suggest that research should be devoted to design for trustworthiness and clear goals (missing in Evernote). Transparency is key in both to allow users to understand the underlying stances of an app and the fit to the users' attitude. A number of limitations of the study exist. First, participants did not actively use the apps, but just looked at screenshots. While we hoped that people would still gain insights about the functionalities and strategies of the apps, we are aware that the actual user experience is different. This maybe one reason for the overall poor mean ratings of the apps and especially Evernote, which is a harder to understand due to its openness and less directive functions. Second, all apps had different styles in user interface design and, we cannot be sure how strong the influence of visual appearance on the ratings of the PTAM was and in how far it may have overruled the aspect of which strategies apps employed (although participants were aware of this study goal). Third, the results presented here are based on a limited number of participants. Fourth, some constructs, especially of the TIPI and some of the PVQ

had low Cronbach's alpha raising the question whether participants understood the questionnaires as intended.

The exploration presented here is only a small step of a larger research agenda to understand values in design, in particular, the relationships between users' values and explicitly supported values (as represented by the stances that apps are designed with). Given the preliminary results we will redefine the study set-up to a simpler version that clearly shows the different strategies of each app and does not add confounding factors such as the design of the user interface. Next, we will conduct a longer-term study in which people actively use the apps in everyday lives. Users will then be sampled based on their values and personality profiles and try both an app matching this profile and one that does not in order to evaluate the differences in user experience. Concluding, we hope that others will follow up on this research direction in order to improve the designs of apps for health behavior change.

References

[1] H. Dubberly, R. Mehta, S. Evenson, and P. Pangaro, *Reframing health to embrace design of our own well-being,* interactions **17**, 56 (2010).

[2] W. H. Organization, *Food and Agricultural Organization. Diet, Nutrition and the Prevention of Chronic Diseases: Report of a Joint WHO/FAO Expert Consultation (Report 916)*, Tech. Rep. (World Health Organization, 2003).

[3] B. Fogg, *Persuasive technology: Using technology to change what we think and do* (San Francisco: Morgan Kaufmann Publishers, 2003).

[4] L. P. Simons, J. F. Hampe, and N. A. Guldemond, *Designing healthy living support: mobile applications added to hybrid (e) coach solution,* Health and Technology **3**, 85 (2013).

[5] J. Davis, *Design methods for ethical persuasive computing,* in *Proceedings of the 4th International Conference on Persuasive Technology*, Persuasive '09 (ACM, New York, NY, USA, 2009) pp. 6:1–6:8.

[6] E. P. Baumer, S. J. Katz, J. E. Freeman, P. Adams, A. L. Gonzales, J. Pollak, D. Retelny, J. Niederdeppe, C. M. Olson, and G. K. Gay, *Prescriptive persuasion and open-ended social awareness: Expanding the design space of mobile health,* in *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work*, CSCW '12 (ACM, New York, NY, USA, 2012) pp. 475–484.

[7] J. Davis, *From ethics to values in the design of mobile pinc,* in *Proceedings of the 2nd International Workshop on Persuasion, Infiuence, Nudge & Coercion Through Mobile Devices co-located with the ACM CHI Conference on Human Factors in Computing Systems, CHI2011* (Citeseer, 2011) pp. 27–30.

[8] A. Borning and M. Muller, *Next steps for value sensitive design,* in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12 (ACM, New York, NY, USA, 2012) pp. 1125–1134.

7

[9] I. Li, A. Dey, and J. Forlizzi, *A stage-based model of personal informatics systems,* in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10 (ACM, New York, NY, USA, 2010) pp. 557–566.

[10] H. Oinas-Kukkonen, *A foundation for the study of behavior change support systems,* Personal Ubiquitous Comput. **17**, 1223 (2013).

[11] S. H. Schwartz, G. Melech, A. Lehmann, S. Burgess, M. Harris, and V. Owens, *Extending the cross-cultural validity of the theory of basic human values with a different method of measurement,* Journal of Cross-Cultural Psychology **32**, 519 (2001), http://jcc.sagepub.com/content/32/5/519.full.pdf+html .

[12] S. Roccas, L. Sagiv, S. H. Schwartz, and A. Knafo, *The big five personality factors and personal values,* Personality and Social Psychology Bulletin **28**, 789 (2002), http://psp.sagepub.com/content/28/6/789.full.pdf+html .

[13] S. D. Gosling, P. J. Rentfrow, and W. B. S. Jr., *A very brief measure of the big-five personality domains,* Journal of Research in Personality **37**, 504 (2003).

[14] C. K. L. Or and B.-T. Karsh, *The patient technology acceptance model (ptam) for homecare patients with chronic illness,* Proceedings of the Human Factors and Ergonomics Society Annual Meeting **50**, 989 (2006), http://pro.sagepub.com/content/50/10/989.full.pdf+html .

7

8

Conclusion

This chapter summarizes the work described in this thesis, presents answers to the research questions posed in Chapter 1, and reflects on this work, the field to which it contributes and directions for future work.

8.1. Research contributions

Technologies change the environments into which they are introduced. They change the way people do things, and make it possible for people to do new things. Whether intentionally or unintentionally, these changes frequently have ethical implications. Many of technologies' possibilities are conceived and shaped during design. In design processes designers set the scope for the design, discover and define the problem to be solved, specify functions and use plans, decide on functions to include and functions to omit. If technology's capabilities influence its ethical implications, and these capabilities are shaped through conscious design decisions and deliberation, then we can make the case that the design process influences a technology's eventual impact on ethically relevant issues.

Technology research and practice are concerned with what we can do and will be able to do, while ethics is concerned with what we should do. In design processes that focus on technical innovation, much of the attention is directed toward tackling technical challenges rather than addressing its ethical issues. The issues might not be apparent, or designers might lack the time and resources to address such issues. Addressing these issues is not generally part of the process of designing a technology.

Design processes consists of different phases. Values can play a role in several of these phases. Decisions with ethical implications can be made in several of these phases. Though it's a matter of discussion whether the design process proceeds from phase to phase in a strict sequence, the act of designing an artifact usually does involve most of the activities that fall within these phases. Often the process starts off with the customer's wishes, which are translated into (functional) requirements, which are translated to design specifications, which is translated to a blueprint, which is used to implement or manufacture the eventual artifact.

While designers might be aware of some of the ethical implications of the technologies they design, they often lack the training or familiarity with techniques to deal with such issues. VSD offers such techniques. VSD offers a range of methods that target various phases of the design process that were described above. VSD makes a distinction between conceptual, empirical and technical investigations. Work on VSD suggests that the methods and techniques developed within the framework (can) be applied iteratively and integratively throughout the design process. Much of the work on VSD describes methods and techniques to apply in various phases. Some of the work employs multiple techniques.

Our work addresses several of the steps that generally make up a design process, emphasizing 'earlier' stages of design during which customer needs are elicited, design goals are specified, existing solutions are examined for inspiration, requirements are specified. Where much of the work on VSD is aimed at the HCI community (cite number of VSD works that were published at CHI and in HCI journals), we attempted to target communities that focus on the design of software and other computational artifacts as well, where human values are not such a familiar concept.

In Chapter 2, we set out to study the relationship between technology and explicitly supported values – values that designers explicitly want to support in their application, which are subjected to a principled analysis of arguments for their inclu-

sion [1], such as privacy-preserving automatic fall detection or intelligent monitoring systems for independent living. We surveyed which values are explicitly supported in an application domain, pervasive computing for elderly care. Furthermore, we analyzed how the constituent technologies of this application domain support those values. The study sought to answer the following questions.

Research question 1 Which values are explicitly supported by Pervasive Computing applications for elderly care? How do this application domain's constituent technologies achieve that support?

Methodologically, this chapter makes three main contributions. First, it turns to applications' descriptions of design goals and problem statements to identify the values they support explicitly. Second, it presents a taxonomy which relates the application domain's constituent technologies to the explicitly supported values. Third, it presents a technique to analyze how constituent technologies are designed to support explicitly supported values.

The main empirical findings of this chapter provide insight into which values are more or less frequently explicitly supported in pervasive computing applications for elderly care. We found that values related to safety and to freedom were explicitly supported most frequently. Both occurred in 19 out of 37 works included in the analysis. Values related to privacy followed closely, being supported in 15 works. The physical well-being group of values was addressed in 9 works included in the analysis, and social well-being in 7 works. Responsibility was the least-supported value in the works we analyzed, occurring in 3 works.

We also found that, in the articles we surveyed, sensing, modeling, and decision making were discussed most frequently (by half of the articles we surveyed or more). The most frequently occurring combinations of values and technologies were sensing with safety, freedom and privacy; modeling with safety and freedom; and decision making with safety, freedom and privacy.

Our analysis revealed that the key values supported in this domain, physical well-being, freedom, privacy and safety, are affected in different ways by sensing, modeling, activity recognition, spatiotemporal reasoning and decision making. For each of these values, sensing technologies are essential to applications' ability to support the value. Subsequent components each expand the applications' support for the value. Each of the components also has the potential to adversely affect the value in question. This happens most frequently for privacy.

The results in this chapter can help identify which values are being addressed in this domain, and which are receiving relatively little attention. The relationships we analyzed between technologies and values in this domain can help designers and researchers understand the potential beneficial and adverse effects certain technologies can have on values, and make informed decisions about incorporating these technologies in their applications.

Chapter 2 demonstrated that pervasive computing projects in the elderly care domain often have aims that explicitly include support for values such as independence and well-being. Many of these values are supported by a variety of applications. The work in Chapter 2 revealed recurrent links between components of

8

pervasive computing systems for elderly care and explicitly supported values. In Chapter 3, we studied how to use this knowledge of recurrent explicit support for values as a means of inspiring and informing design for values. The chapter sought to answer the following question.

Research question 2 How can recurrent value issues surrounding pervasive computing applications for health care be identified, analyzed and used to guide design?

This paper makes four main methodological contributions. First, it proposes a technique to identify the values that selected existing technologies explicitly support, by examining descriptions of these technologies for value statements. Second, it presents a technique to identify the relationship between explicitly supported values and the systems that support them. Third, it proposes a technique to discover design patters from the knowledge gathered in the first two steps. Finally, the paper presents a number of design patterns we identified in a body of literature using our technique.

Empirically, this chapter provides insight into which values are explicitly supported in pervasive computing applications for elderly care and how. These contributions are a first step towards helping designers that are not familiar with the roles values can play in design to account for values and learn about the relationships between values and technologies, and incorporate this knowledge into their design process. We have contributed a way for explicitly supported values in existing technologies to guide design early in the design process or guide ideation.

Chapter 4 studied how technologies can affect human values in unforeseen or unintended ways. In particular, the chapter analyzed how pervasive computing technology for elderly care can affect the value of responsibility, a relatively unaddressed value in this domain, in unforeseen and/or unintended ways through the types of knowledge this technology makes available. Chapter 4 sought to answer the following research questions.

8

Research question 3 How do pervasive computing technologies for elderly care affect values that are not explicitly supported? How can such effects be identified, analyzed and made explicit?

This chapter provides insight into the relationship between responsibility and knowledge, and examines how the introduction of knowledge-based systems to a context introduces and changes existing responsibilities and introduces new ones. Furthermore, it identifies the direct and indirect stakeholders in the elderly care domain, and examines how the value of responsibility can be defined for these groups. It analyzes how the introduction of pervasive computing applications in this domain changes and introduces responsibilities for these stakeholders. We captured our analysis in semi-formal language, making it amenable as a starting point to be included in more formal requirements specification.

This chapter focuses on stakeholder values and their role early in the design process, when a technology is yet to be designed. It studies ways of discovering

these values and incorporating them into further steps in the design process. The chapter addresses the following questions.

Research question 4 How can values be elicited within a Requirements Engineering process? How can elicited values be translated into design requirements?

This chapter provides insight into the the extent to which current methods can help discover values and incorporate them in a Requirements Engineering process. We examined techniques developed within Value Sensitive Design that focus on value elicitation to assess to what extent these could be used in Requirements Engineering processes. We found that, though these techniques help consider stakeholders, their values and a system's (long-term) implications for values, they do not guide the designer or requirements engineer in specifying system functions and qualities to support the values in question.We also examined techniques developed within Requirements Engineering to deal with with values. We found that, upon closer examination, these techniques focus on emotions rather than values. Where they do help elicit values, they do so for other aims than discovering or specifying requirements to support values, leaving this an open challenge.

The methodological contributions of this chapter include a method to elicit values from stakeholders and translate these into design requirements, in a way that would fit into existing design processes as much as possible, and that could easily be translated into requirements to ensure that the elicited values are dealt with adequately in design. It builds on techniques developed within the VSD framework to guide identification of direct and indirect stakeholders. The method introduces a technique to identify and analyze these stakeholders' values, and capture stakeholders' definitions of these values through examples concrete situations. The method also introduces a technique to distill stakeholder needs from these concrete situations and, based on these stakeholder needs, creating user stories that include values. Furthermore, this chapter contributes a workshop format in which to apply the proposed method.

Application of this method in a case study yielded a number of insights. First, the participants seemed to understand the main ideas of VSD and what was expected of them in the workshop. Second, the participants were able to accomplish all steps in the workshop. Third, the workshop evoked discussions and information exchanges among the participants, and that seemed to contribute in developing a common view on the goals of the IQmulus projects. A potential drawback of the technique is that it requires a considerable amount of time. In this workshop, 13 stakeholders were identified, but in the total duration of the workshop, 4 hours, only 3 stakeholders were analyzed.

These contributions brings the concept of (stakeholder) values to Requirements Engineering, and helps translate elicited values to requirements, helping bridge the gap between VSD and Requirements Engineering.

This chapter studies the issue of making the influence of values on the design explicit during all design phases, and the use of (semi-)formal modeling methods to inform technical investigations of values. The chapter addresses the following questions.

8

Research question 5 What concepts in software design could potentially capture value knowledge? What are the similarities and differences between these concepts and values? What should a design process adhere to if it is to incorporate knowledge about values?

This chapter contributes a number of insights. We examined to what extent existing Agent-Oriented Software Engineering methods deal with values or similar concepts, and which concepts they use in doing so. Before conducting this analysis, we defined values and differentiated the concept from similar concepts, such as laws, rules, goals, norms, and standards, arguing that values have a special status that they gain from people's attitudes and responses toward them. Considering the status of moral values' and the importance of dealing with them in design, we formulated six principles to avoid the negative consequences of violating values and to promote value support as much as possible.

1. The values of all stakeholders including designers/developers should be elicited in as far as relevant for the system under design.

2. Conflicts between values of the designers and those of the stakeholders need to be discussed with those who issued the order for the system.

3. To account for the relevant values, to the relevant values need to be instantiated explicitly throughout the design process.

4. Design decisions can and need to be justified and evaluated in terms of explicit (instantiations of) stakeholders' values.

5. Conflicts between values need to be made clear and addressed in cooperation with the stakeholders.

We proceeded to analyze how existing approaches deal with values or similar concepts, focusing on our proposed principles. We identified several potential concepts, including non-functional requirements, quality attributes, soft constraints, and soft-goals. We found that soft-goals, as used in the Tropos methodology, could potentially deal with values. To examine Tropos' potential to deal with values by using soft-goals, we conducted a case study in which we applied the Tropos method to design a conference management system (an example often used to compare Agent-Oriented Software Engineering methods). Our case study revealed that Tropos can satisfy the first, fifth and sixth principles to some extent, and can satisfy the second and fourth if we consider values to be goals, but cannot satisfy the third. We argued that this is the very reason Tropos cannot adhere to the principles, as adherence would require representing values as soft-goals. But values, we argued, should not be treated as soft-goals due to fundamental differences between the concepts.

To deal with these issues, we proposed complementing Tropos with a separate, first-class entity to capture values. This entity would allow the designer to explicitly represent values throughout the design process. It would also make values concrete enough to operationalize them, and would help expose and address conflicts between them.

In Chapters 2, 3 and 4 we examined how to identify explicitly supported values for an application domain, and how to analyze how values other than those explicitly supported might be affected by an application. Another important source of values to consider is the values of stakeholders of a technology. Chapter 4 does this to some extent, but analyzes in which ways a certain value could be relevant to different stakeholder groups and how it might be affected by an application. In Chapter 7, we turn towards the stakeholders themselves to identify which values might be important and how these values might be affected.

Beginning with the observation that Pervasive Computing applications that share a goal to improve users' personal well-being by helping them adopt and maintain healthy eating habits take very different approaches to meet this goal. Based on open interviews with these applications' designers, we hypothesized that the applications' different approaches were due, at least in part, to the designers' views on users' autonomy. In other words, the different approaches to the same goal could be explained by designer values. We hypothesized that direct stakeholders' (that is, users) values would affect their preferences for the different applications, so that they would be more likely to adopt applications that embody values that align with their own. More specifically, we hypothesized that people who gave higher priority to values related to autonomy would prefer applications that gave users more autonomy.

Research question 6 How do designer values and stakeholder values affect stakeholders' acceptance of Pervasive Computing applications for personal well-being? Do similarities and differences between designer and stakeholder values impact how stakeholders experience the technology?

To examine this, we needed to discover stakeholder values and examine the same stakeholders' preferences for applications that aim for the same goals but embody different values. We designed a study (that could be administered remotely) in which we would elicit participants' values, expose them to a selection of applications, and assess their acceptance of each application. We used the 21-item Portrait Value Questionnaire to elicit stakeholder value preferences. We included the Ten-Item Personality Inventory to check whether personality had an effect on application preferences.

To examine application preferences, we selected three existing smartphone applications related to eating. Two of the applications aim to help users adopt and maintain healthy eating habits — one by prescribing healthy eating behavior, the other by helping users track and reflect on their eating behavior. The third application was neutral with regard to healthy eating habits, allowing users to capture images and text about food according to their own needs or desires. The study's remote setup precluded letting participants interact with the applications in a controlled setting. So, rather than have participants use the actual applications, we created storyboards that allowed participants to familiarize themselves with the applications' functionality. To assess participants' preferences for these applications, we assessed their acceptance of each application individually using a version of the Technology Acceptance Model aimed at mobile health applications called the Patient

Technology Acceptance Model. We assumed that differences in intention to use the different applications (which the PTAM measures) reflected preferences for these applications.

Our results revealed significant differences in how the applications were rated. However, we found few correlations between values and personality factors, and the intention to use an application. We did find that the intention to use Evernote, the least neutral application, was rated low all in all groups of people. This may be based on the low perceived usefulness. The fact that the value of hedonism was negatively correlated with Evernote suggests that participants did not think the application seemed fun to use. Interestingly, people who preferred values related to power seemed to trust the applications less. Although we cannot draw general conclusions from them, the results suggest that research should be devoted to design for trustworthiness and clear goals. Transparency is key both in helping users understand the stances an application embodies with regard to certain values, and in matching the users' attitude.

8.2. Reflections and future work

When we began the work that makes up this thesis, several people had been working on Value Sensitive Design and related approaches for over a decade. Many theories, methods and techniques had already emerged as the overall framework evolved, which added the options available to designers to engage with stakeholders, discover values, conceptualize values, identify and address value trade-offs, envision the long-term effects of a technology, and generally deal with values in design. Much of this work was conducted in the field of Human-Computer Interaction – of the 154 publications in the ACM Digital Library that appeared between 1996 and 2009 and mention "Value Sensitive Design", 78 were published in HCI focused conference proceedings and journals such as CHI and Interactions. Many of VSD's methods and techniques fall within this scope. At the time, work on VSD had focused less on requirements engineering and software engineering, which are also make important contributions to the design of ICTs. At the same time, human values had not received much attention in those fields.

The work in this thesis contributes to both VSD and requirements engineering. Each of the chapters in this thesis deals with human values. The sources of these values varies. Chapter 2 examined project descriptions or definitions in literature on pervasive computing technology for elderly care to find instances of explicitly expressed values – values that each project aimed to support. In Chapter 3, these explicitly expressed values are also the starting point. Stakeholders are another source of values. The work in Chapter 5 deals with eliciting values from stakeholders in workshops. In Chapter 6, values are elicited in stakeholder interviews. In Chapter 7, online surveys are used to elicit stakeholders' value preferences.

Several of the chapters in this thesis proceed to analyze the uncovered values. Chapter 2 analyzes relationships between identified values, components of the systems that aim to support them, and contexts of use, in order to make these value concepts more concrete and (context-)specific. Chapter 3 analyzes values in problem statements and proposed solutions, and identifies patterns in these. In Chapter

4, a conceptual analysis is performed for the value of responsibility in the context of pervasive computing technology for elderly care. The Value Stories Workshop method described in Chapter 5 analyzes elicited stakeholder values to make them more concrete and translate them into requirements. In Chapter 6, values are introduced as a concept in Tropos models and requirements analysis is performed. Chapter 7 analyzes the relationship between stakeholders' value priorities, different applications and stakeholders' acceptance of those applications.

Finally, most of the chapters in this thesis capture the results of these analysis in a design artifact. In Chapter 2, this is a taxonomy of values, contexts and technologies (in pervasive computing for elderly care). Chapter 3 presents Value-Sensitive Design Patterns. In Chapter 4, the performed analysis is captured in a semi-formal language. Chapter 5 captures analyses performed during workshops in Value Stories, which are a form of high-level requirements. Chapter 6 captures value analysis in Tropos requirements models.

Reflecting on this work, its components appear to fall into three broad categories of activity: identifying and examining sources of values, analyzing uncovered values, and capturing analyses in artifacts. These three general activities form a process, represented in Figure 8.1
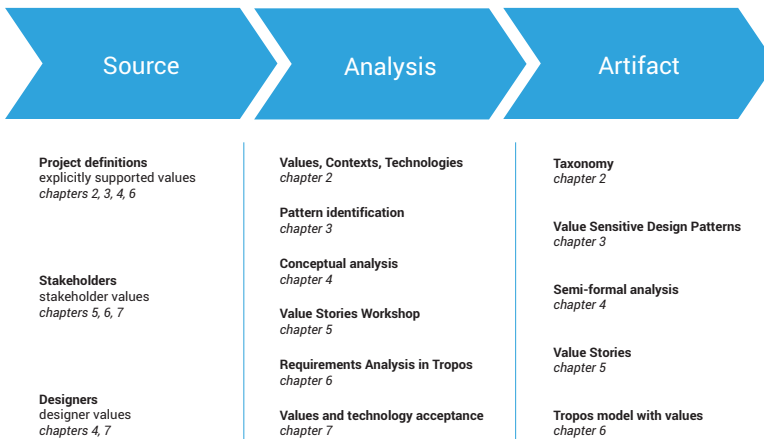
Figure 8.1: Process

These steps resemble the constitutive activities Flanagan and colleagues' methodological framework for incorporating values in the design process: discovery, translation and verification (cf. [2]). Discovery aims to "'discover' the values that are relevant to, inspire or inform a given design project, resulting in a list of values, and bringing into focus what is often implicit in a design project" [2, p.334]. It looks for values in the definition of a project, values that emerge in specifying instrumental design features, designers' values, and users' values. This is similar to the activities in our work that deal with identifying and examining sources of values. The translation activity aims to express the discovered values in system design. It is divided into operationalization, which "involves defining, or articulating values in concrete

terms" [2, p.338], and implementation, in which corresponding design features are specified. Our analyses align with these activities. Finally, verification deals with assessing to what extent target values were successfully implemented in a system. Our work on stakeholder values and technology acceptance has similar aims.

In this thesis, we have followed 5 different paths through this process from source, via analysis, to artifact. Other paths are yet to be explored. For example, our Value Sensitive Design Patterns use explicitly supported values as a source, but another important source to examine is stakeholder values. Similarly, the Value Stories capture stakeholder values. It would be interesting to use Value Stories to elicit and capture designers' values.

A related direction for future work is to study this overall process and the interactions between (the outcomes of) the various activities. Some artifacts that result from the process can serve as the source for further iterations. For example, Value Stories could form the basis for Value-Sensitive Design Patterns.

Future work should also study and evaluate these methods in use in actual design processes, from early phases of design through implementation. It should examine the advantages and difficulties of applying these methods in a design process. Ultimately, it should assess to what extent these methods help incorporate human values in design.

References

[1] A. Borning, B. Friedman, J. Davis, and P. Lin, *Informing public deliberation: Value sensitive design of indicators for a large-scale urban simulation,* in *ECSCW 2005*, edited by H. Gellersen, K. Schmidt, M. Beaudouin-Lafon, and W. Mackay (Springer Netherlands, 2005) pp. 449–468, 10.1007/1-4020-4023-7_23.

[2] M. Flanagan, D. Howe, and H. Nissenbaum, *Embodying values in technology: Theory and practice,* Information technology and moral philosophy , 322 (2008).

8

Acknowledgements

My supervisors made this whole endeavor possible in the first place. Thanks to Catholijn Jonker for challenging me from the onset, and for her interest not only in how my work was going, but how I was doing as well (what she called "meta-conversations"). I am grateful to Jeroen van den Hoven for his enthusiasm and encouragement, and his uncanny ability to come up with spot-on literature suggestions based on my vague descriptions of what I planned to work on next. Koen Hindriks patiently made sure that I stayed on track and that my writing was focused. I won't forget his "what's the storyline?" for as long as I write (and my students probably won't either). Thank you, Koen. Alfons Salden supervised me at Almende BV. Our discussions were always lively, and Alfons' stream of ideas was a note-taking challenge (not to mention the challenge of making sense of my notes after a meeting). I am also grateful to Jan Peter Larsen, my manager at Almende, who helped me get involved in interesting projects with Almende and its partners.

Above, I suggested that my supervisors made this whole endeavor possible in the first place. That's not entirely accurate. Joost Broekens supervised me on my MSc. thesis at Leiden University. He later moved to the Interactive Intelligence group and suggested that I apply for a PhD position there. Joost, thank you for your enthusiasm, ideas and friendship. Our talks by the coffee machines in Leiden and Delft were fun, and it always felt like we were on to something.

The work for this thesis happened at the Interactive Intelligence Group and the Department of Values, Technology and Innovation at Delft University of Technology, and at Almede BV in Rotterdam (with some finishing touches at The Hague University of Applied Sciences). I would like to thank all of my colleagues there for making each of these places pleasant environments. Duco, Freek, Gert-Jan, Joris, and Ted at Almende, thanks for the lunchtime discussions (or debates, whenever Duco was there), the jam sessions and the occasional course in film history. Also, thanks Andries, Anne, Emine, Hans, Judith, and Peet for the support and the chats about research, agents, music technology, travels and a multitude of other things.

Luca and Roberto, my philosophical roommates, thank you for all the laughs and lively discussions in our room, during breaks and over dinner. Engin, monsieur!, it was always fun coming up with cases for the ethics and IT classes we taught, as was organizing a summer school. Best of all were the Lisbon and Paris conferences. Châtelet? Châtelet. Thanks Benham, Christine, David, Francien, Ilse, Neelke, Nicole, Noëmi, and Sabine for the reading groups, teaching, chats, collaboration and general pleasantness.

Most of my time was probably spent at the Interactive Intelligence group. Alina, in you I found someone with similar research interests, with whom to excitedly discuss ideas and collaborate (from initial idea to paper: 5 hours on a Friday afternoon). It was inspiring to work with you, and so much fun to organize workshops

Curriculum Vitæ

Christian Alexander Detweiler

20-12-1980   Born in Morgantown, West Virginia, United States

Education

1993–1999   Grammar School

1999–2005   Bachelor of Arts in Media and Culture Studies
University of Amsterdam

2005–2008   Master of Science in Media Technology
Leiden University
Thesis: Trust in Online Technology
Supervisor: Dr. D.J. Broekens

List of Publications

15. **Detweiler, C. A.** and Hindriks, K. V. , *A Survey of Values, Technologies and Contexts in Pervasive Healthcare*, Pervasive and Mobile Computing, Volume 27, Pages 1-13, (April 2016).

14. Zevenbergen, B. , Mittelstadt, B. , Veliz, C. , **Detweiler, C. A.** , Cath, C., Savulescu, J. and Whittaker, M. , *Philosophy meets internet engineering: Ethics in networked systems research. (GTC workshop outcomes paper).*, Social Science Research Network Working Paper Series, (2015)..

13. Harbers, M., **Detweiler, C. A.**, and Neerincx, M. A., *Embedding Stakeholder Values in the Requirements Engineering Process*, Requirements Engineering: Foundations for Sofware Quality (2015).

12. **Detweiler, C. A.** , Harbers, M. and Hindriks, K. V. , *Value Stories: Putting Values into Requirements Engineering*, Proceedings of the Fourth International Workshop on Creativity in Requirements Engineering at REFSQ (2014).

11. **Detweiler, C. A.** and Hindriks, K. V. , *Value-sensitive design patterns for pervasive health care*, Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012 IEEE International Conference on Pervasive Computing (2012).

10. **Detweiler, C. A.** , Dechesne, F. , Hindriks, K. V. and Jonker, C. M. , *Ambient Intelligence Implies Responsibility*, Ambient Intelligence and Smart Environments **12,** (2012).

9. Pommeranz, A. , **Detweiler, C. A.** Wiggers, P. and Jonker, C. M. , *Elicitation of situated values: need for tools to help stakeholders and designers to reflect and communicate*, Ethics and Information Technology **14,** 4 (2012).

8. **Detweiler, C. A.** and Pommeranz, A. , *Go slow and beware! A call for reflection on our computational surroundings* , Slow Technology DIS 2012 Workshop (2012).

7. **Detweiler, C. A.** , Pommeranz, A. and Stark, C. L. , *Methods to account for values in human-centered computing* , CHI'12 Extended Abstracts on Human Factors in Computing Systems (2012).

6. O'Kane, A. , **Detweiler, C. A.** and Pommeranz, A. , *Designing and evaluating for trust: a perspective from the new practitioners*, 1st International Workshop on Values in Design–Building Bridges between RE, HCI and Ethics, (2011).

5. **Detweiler, C. A.** , Pommeranz, A. , van den Hoven, M. J. and Nissenbaum, H. , *Values in design-building bridges between RE, HCI and ethics*, Human-Computer Interaction– INTERACT 2011 (2011).

4. Pommeranz, A. , **Detweiler, C. A.** Wiggers, P. and Jonker, C. M. , *Self-reflection on personal values to support value-sensitive design*, Proceedings of the 25th BCS Conference on Human-Computer Interaction (2011).

3. **Detweiler, C. A.** , Hindriks, K. V. and Jonker, C. M. , *Principles for value-sensitive agent-oriented software engineering*, Agent-Oriented Software Engineering XI (2011).

2. **Detweiler, C. A.** , Hindriks, K. V. and Jonker, C. M. , *A Value-Sensitive Approach to Agent-Oriented Software Engineering*, Workshop 8 The Eleventh International Workshop on Agent Oriented Software Engineering AOSE 2010 (2010).

1. **Detweiler, C. A.** and Broekens, J., *Trust in Online Technology: Towards Practical Guidelines Based on Experimentally Verified Theory*, Human-Computer Interaction. Ambient, Ubiquitous and Intelligent Interaction **5612** (2009).

SIKS Dissertation Series

```
====
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 (UT)
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 (UT)
        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 (UT)
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 Mihajlović (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 (UT)
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 Jovanović (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 (UT)
    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 Henriëtte 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 (UT)
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 (UT)
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)

2009-37 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 (UT)
    Multinomial Language Learning: Investigations into the Geometry of Language

2009-41 Igor Berezhnyy (UT)
    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 (UT)
    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 Politiële 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 Henriëtte 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 (UT)
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 (UT)
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 (UT)
      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

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 (UT)
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
====

2012-01 Terry Kakeeto (UT)
Relationship Marketing for SMEs in Uganda

2012-02 Muhammad Umair(VU)
Adaptivity, emotion, and Rationality in Human and Ambient Agent Models

2012-03 Adam Vanya (VU)
Supporting Architecture Evolution by Mining Software Repositories

2012-04 Jurriaan Souer (UU)
Development of Content Management System-based Web Applications

2012-05 Marijn Plomp (UU)
Maturing Interorganisational Information Systems

2012-06 Wolfgang Reinhardt (OU)
Awareness Support for Knowledge Workers in Research Networks

2012-07 Rianne van Lambalgen (VU)
When the Going Gets Tough: Exploring Agent-based Models of Human Performance under Demanding Conditions

2012-08 Gerben de Vries (UVA)
Kernel Methods for Vessel Trajectories

2012-09 Ricardo Neisse (UT)
Trust and Privacy Management Support for Context-Aware Service Platforms

2012-10 David Smits (TUE)
Towards a Generic Distributed Adaptive Hypermedia Environment

2012-11 J.C.B. Rantham Prabhakara (TUE)
Process Mining in the Large: Preprocessing, Discovery, and Diagnostics

2012-12 Kees van der Sluijs (TUE)
Model Driven Design and Data Integration in Semantic Web Information Systems

2012-13 Suleman Shahid (UT)
Fun and Face: Exploring non-verbal expressions of emotion during playful interactions

2012-14 Evgeny Knutov(TUE)
Generic Adaptation Framework for Unifying Adaptive Web-based Systems

2012-15 Natalie van der Wal (VU)
Social Agents. Agent-Based Modelling of Integrated Internal and Social Dynamics of Cognitive and Affective Processes

2012-16 Fiemke Both (VU)
Helping people by understanding them - Ambient Agents supporting task execution and depression treatment

2012-17 Amal Elgammal (UT)

Towards a Comprehensive Framework for Business Process Compliance

2012-18 Eltjo Poort (VU)
What's Next? Operational Support for Business Process Execution

2012-19 Helen Schonenberg (TUE)
What's Next? Operational Support for Business Process Execution

2012-20 Ali Bahramisharif (RUN)
Covert Visual Spatial Attention, a Robust Paradigm for Brain-Computer Interfacing

2012-21 Roberto Cornacchia (TUD)
Querying Sparse Matrices for Information Retrieval

2012-22 Thijs Vis (UT)
Intelligence, politie en veiligheidsdienst: verenigbare grootheden?

2012-23 Christian Muehl (UT)
Toward Affective Brain-Computer Interfaces: Exploring the Neurophysiology of Affect during Human Media Interaction

2012-24 Laurens van der Werff (UT)
Evaluation of Noisy Transcripts for Spoken Document Retrieval

2012-25 Silja Eckartz (UT)
Managing the Business Case Development in Inter-Organizational IT Projects: A Methodology and its Application

2012-26 Emile de Maat (UVA)
Making Sense of Legal Text

2012-27 Hayrettin Gurkok (UT)
Mind the Sheep! User Experience Evaluation & Brain-Computer Interface Games

2012-28 Nancy Pascall (UT)
Engendering Technology Empowering Women

2012-29 Almer Tigelaar (UT)
Peer-to-Peer Information Retrieval

2012-30 Alina Pommeranz (TUD)
Designing Human-Centered Systems for Reflective Decision Making

2012-31 Emily Bagarukayo (RUN)
A Learning by Construction Approach for Higher Order Cognitive Skills Improvement, Building Capacity and Infrastructure

2012-32 Wietske Visser (TUD)
Qualitative multi-criteria preference representation and reasoning

2012-33 Rory Sie (OUN)
Coalitions in Cooperation Networks (COCOON)

2012-34 Pavol Jancura (RUN)
Evolutionary analysis in PPI networks and applications

2012-35 Evert Haasdijk (VU)
Never Too Old To Learn – On-line Evolution of Controllers in Swarm- and Modular Robotics

2012-36 Denis Ssebugwawo (RUN)
Analysis and Evaluation of Collaborative Modeling Processes

2012-37 Agnes Nakakawa (RUN)
A Collaboration Process for Enterprise Architecture Creation

2012-38 Selmar Smit (VU)
Parameter Tuning and Scientific Testing in Evolutionary Algorithms

2012-39 Hassan Fatemi (UT)
Risk-aware design of value and coordination networks

2012-40 Agus Gunawan (UT)
Information Access for SMEs in Indonesia

2012-41 Sebastian Kelle (OU)
Game Design Patterns for Learning

2012-42 Dominique Verpoorten (OU)
Reflection Amplifiers in self-regulated Learning

2012-43 Withdrawn

2012-44 Anna Tordai (VU)
On Combining Alignment Techniques

2012-45 Benedikt Kratz (UT)
A Model and Language for Business-aware Transactions

2012-46 Simon Carter (UVA)
Exploration and Exploitation of Multilingual Data for Statistical Machine Translation

2012-47 Manos Tsagkias (UVA)
Mining Social Media: Tracking Content and Predicting Behavior

2012-48 Jorn Bakker (TUE)
Handling Abrupt Changes in Evolving Time-series Data

2012-49 Michael Kaisers (UM)
Learning against Learning - Evolutionary dynamics of reinforcement learning algorithms in strategic interactions

2012-50 Steven van Kervel (TUD)
Ontologogy driven Enterprise Information Systems Engineering

2012-51 Jeroen de Jong (TUD)
Heuristics in Dynamic Sceduling; a practical framework with a case study in elevator dispatching

====
2013
====

2013-01 Viorel Milea (EUR)
News Analytics for Financial Decision Support

2013-02 Erietta Liarou (CWI)
MonetDB/DataCell: Leveraging the Column-store Database Technology for
Efficient and Scalable Stream Processing

2013-03 Szymon Klarman (VU)
Reasoning with Contexts in Description Logics

2013-04 Chetan Yadati(TUD)
Coordinating autonomous planning and scheduling

2013-05 Dulce Pumareja (UT)
Groupware Requirements Evolutions Patterns

2013-06 Romulo Goncalves(CWI)
The Data Cyclotron: Juggling Data and Queries for a Data Warehouse Audience

2013-07 Giel van Lankveld (UT)
Quantifying Individual Player Differences

2013-08 Robbert-Jan Merk(VU)
Making enemies: cognitive modeling for opponent agents in fighter pilot simulators

2013-09 Fabio Gori (RUN)
    Metagenomic Data Analysis: Computational Methods and Applications

2013-10 Jeewanie Jayasinghe Arachchige(UT)
    A Unified Modeling Framework for Service Design.

2013-11 Evangelos Pournaras(TUD)
    Multi-level Reconfigurable Self-organization in Overlay Services

2013-12 Marian Razavian(VU)
    Knowledge-driven Migration to Services

2013-13 Mohammad Safiri(UT)
    Service Tailoring: User-centric creation of integrated IT-based homecare services to
    support independent living of elderly

2013-14 Jafar Tanha (UVA)
    Ensemble Approaches to Semi-Supervised Learning Learning

2013-15 Daniel Hennes (UM)
    Multiagent Learning - Dynamic Games and Applications

2013-16 Eric Kok (UU)
    Exploring the practical benefits of argumentation in multi-agent deliberation

2013-17 Koen Kok (VU)
    The PowerMatcher: Smart Coordination for the Smart Electricity Grid

2013-18 Jeroen Janssens (UT)
    Outlier Selection and One-Class Classification

2013-19 Renze Steenhuizen (TUD)
    Coordinated Multi-Agent Planning and Scheduling

2013-20 Katja Hofmann (UvA)
    Fast and Reliable Online Learning to Rank for Information Retrieval

2013-21 Sander Wubben (UT)
    Text-to-text generation by monolingual machine translation

2013-22 Tom Claassen (RUN)
    Causal Discovery and Logic

2013-23 Patricio de Alencar Silva(UT)
    Value Activity Monitoring

2013-24 Haitham Bou Ammar (UM)
    Automated Transfer in Reinforcement Learning

2013-25 Agnieszka Anna Latoszek-Berendsen (UM)
    Intention-based Decision Support.
    A new way of representing and implementing clinical guidelines in a Decision Support System

2013-26 Alireza Zarghami (UT)
    Architectural Support for Dynamic Homecare Service Provisioning

2013-27 Mohammad Huq (UT)
    Inference-based Framework Managing Data Provenance

2013-28 Frans van der Sluis (UT)
    When Complexity becomes Interesting: An Inquiry into the Information eXperience

2013-29 Iwan de Kok (UT)
    Listening Heads

2013-30 Joyce Nakatumba (TUE)
    Resource-Aware Business Process Management: Analysis and Support

2013-31 Dinh Khoa Nguyen (UT)
Blueprint Model and Language for Engineering Cloud Applications

2013-32 Kamakshi Rajagopal (OUN)
Networking For Learning; The role of Networking in a Lifelong Learner's Professional Development

2013-33 Qi Gao (TUD)
User Modeling and Personalization in the Microblogging Sphere

2013-34 Kien Tjin-Kam-Jet (UT)
Distributed Deep Web Search

2013-35 Abdallah El Ali (UvA)
Minimal Mobile Human Computer Interaction

2013-36 Than Lam Hoang (TUe)
Pattern Mining in Data Streams

2013-37 Dirk Börner (OUN)
Ambient Learning Displays

2013-38 Eelco den Heijer (VU)
Autonomous Evolutionary Art

2013-39 Joop de Jong (TUD)
A Method for Enterprise Ontology based Design of Enterprise Information Systems

2013-40 Pim Nijssen (UM)
Monte-Carlo Tree Search for Multi-Player Games

2013-41 Jochem Liem (UVA)
Supporting the Conceptual Modelling of Dynamic Systems: A Knowledge Engineering Perspective on Qualitative Reasoning

2013-42 Léon Planken (TUD)
Algorithms for Simple Temporal Reasoning

2013-43 Marc Bron (UVA)
Exploration and Contextualization through Interaction and Concepts

====
2014
====

2014-01 Nicola Barile (UU)
Studies in Learning Monotone Models from Data

2014-02 Fiona Tuliyano (RUN)
Combining System Dynamics with a Domain Modeling Method

2014-03 Sergio Raul Duarte Torres (UT)
Information Retrieval for Children: Search Behavior and Solutions

2014-04 Hanna Jochmann-Mannak (UT)
Websites for children: search strategies and interface design -
Three studies on children's search performance and evaluation

2014-05 Jurriaan van Reijsen (UU)
Knowledge Perspectives on Advancing Dynamic Capability

2014-06 Damian Tamburri (VU)
Supporting Networked Software Development

2014-07 Arya Adriansyah (TUE)
Aligning Observed and Modeled Behavior

2014-08 Samur Araujo (TUD)
Data Integration over Distributed and Heterogeneous Data Endpoints

2014-09 Philip Jackson (UT)
Toward Human-Level Artificial Intelligence: Representation and Computation of Meaning in Natural Language

2014-10 Ivan Salvador Razo Zapata (VU)
Service Value Networks

2014-11 Janneke van der Zwaan (TUD)
An Empathic Virtual Buddy for Social Support

2014-12 Willem van Willigen (VU)
Look Ma, No Hands: Aspects of Autonomous Vehicle Control

2014-13 Arlette van Wissen (VU)
Agent-Based Support for Behavior Change: Models and Applications in Health and Safety Domains

2014-14 Yangyang Shi (TUD)
Language Models With Meta-information

2014-15 Natalya Mogles (VU)
Agent-Based Analysis and Support of Human Functioning in Complex Socio-Technical Systems:
Applications in Safety and Healthcare

2014-16 Krystyna Milian (VU)
Supporting trial recruitment and design by automatically interpreting eligibility criteria

2014-17 Kathrin Dentler (VU)
Computing healthcare quality indicators automatically:
Secondary Use of Patient Data and Semantic Interoperability

2014-18 Mattijs Ghijsen (UVA)
Methods and Models for the Design and Study of Dynamic Agent Organizations

2014-19 Vinicius Ramos (TUE)
Adaptive Hypermedia Courses: Qualitative and Quantitative Evaluation and Tool Support

2014-20 Mena Habib (UT)
Named Entity Extraction and Disambiguation for Informal Text: The Missing Link

2014-21 Kassidy Clark (TUD)
Negotiation and Monitoring in Open Environments

2014-22 Marieke Peeters (UU)
Personalized Educational Games - Developing agent-supported scenario-based training

2014-23 Eleftherios Sidirourgos (UvA/CWI)
Space Efficient Indexes for the Big Data Era

2014-24 Davide Ceolin (VU)
Trusting Semi-structured Web Data

2014-25 Martijn Lappenschaar (RUN)
New network models for the analysis of disease interaction

2014-26 Tim Baarslag (TUD)
What to Bid and When to Stop

2014-27 Rui Jorge Almeida (EUR)
Conditional Density Models Integrating Fuzzy and Probabilistic Representations of Uncertainty

2014-28 Anna Chmielowiec (VU)
Decentralized k-Clique Matching

2014-29 Jaap Kabbedijk (UU)
Variability in Multi-Tenant Enterprise Software

2014-30 Peter de Cock (UT)
Anticipating Criminal Behaviour

2014-31 Leo van Moergestel (UU)
Agent Technology in Agile Multiparallel Manufacturing and Product Support

2014-32 Naser Ayat (UvA)
On Entity Resolution in Probabilistic Data

2014-33 Tesfa Tegegne (RUN)
Service Discovery in eHealth

2014-34 Christina Manteli(VU)
The Effect of Governance in Global Software Development: Analyzing Transactive Memory Systems.

2014-35 Joost van Ooijen (UU)
Cognitive Agents in Virtual Worlds: A Middleware Design Approach

2014-36 Joos Buijs (TUE)
Flexible Evolutionary Algorithms for Mining Structured Process Models

2014-37 Maral Dadvar (UT)
Experts and Machines United Against Cyberbullying

2014-38 Danny Plass-Oude Bos (UT)
Making brain-computer interfaces better: improving usability through post-processing.

2014-39 Jasmina Maric (UT)
Web Communities, Immigration, and Social Capital

2014-40 Walter Omona (RUN)
A Framework for Knowledge Management Using ICT in Higher Education

2014-41 Frederic Hogenboom (EUR)
Automated Detection of Financial Events in News Text

2014-42 Carsten Eijckhof (CWI/TUD)
Contextual Multidimensional Relevance Models

2014-43 Kevin Vlaanderen (UU)
Supporting Process Improvement using Method Increments

2014-44 Paulien Meesters (UT)
Intelligent Blauw: Intelligence-gestuurde politiezorg in gebiedsgebonden eenheden.

2014-45 Birgit Schmitz (OUN)
Mobile Games for Learning: A Pattern-Based Approach

2014-46 Ke Tao (TUD)
Social Web Data Analytics: Relevance, Redundancy, Diversity

2014-47 Shangsong Liang (UVA)
Fusion and Diversification in Information Retrieval

====
2015
====

2015-01 Niels Netten (UvA)
Machine Learning for Relevance of Information in Crisis Response

2015-02 Faiza Bukhsh (UT)
Smart auditing: Innovative Compliance Checking in Customs Controls

2015-03 Twan van Laarhoven (RUN)
      Machine learning for network data

2015-04 Howard Spoelstra (OUN)
      Collaborations in Open Learning Environments

2015-05 Christoph Bösch(UT)
      Cryptographically Enforced Search Pattern Hiding

2015-06 Farideh Heidari (TUD)
      Business Process Quality Computation - Computing Non-Functional Requirements to Improve Business Processes

2015-07 Maria-Hendrike Peetz(UvA)
      Time-Aware Online Reputation Analysis

2015-08 Jie Jiang (TUD)
      Organizational Compliance: An agent-based model for designing and evaluating organizational interactions

2015-09 Randy Klaassen(UT)
      HCI Perspectives on Behavior Change Support Systems

2015-10 Henry Hermans (OUN)
      OpenU: design of an integrated system to support lifelong learning

2015-11 Yongming Luo(TUE)
      Designing algorithms for big graph datasets: A study of computing bisimulation and joins

2015-12 Julie M. Birkholz (VU)
      Modi Operandi of Social Network Dynamics: The Effect of Context on Scientific Collaboration Networks

2015-13 Giuseppe Procaccianti(VU)
      Energy-Efficient Software

2015-14 Bart van Straalen (UT)
      A cognitive approach to modeling bad news conversations

2015-15 Klaas Andries de Graaf (VU)
      Ontology-based Software Architecture Documentation

2015-16 Changyun Wei (UT)
      Cognitive Coordination for Cooperative Multi-Robot Teamwork

2015-17 André van Cleeff (UT)
      Physical and Digital Security Mechanisms: Properties, Combinations and Trade-offs

2015-18 Holger Pirk (CWI)
      Waste Not, Want Not! - Managing Relational Data in Asymmetric Memories

2015-19 Bernardo Tabuenca (OUN)
      Ubiquitous Technology for Lifelong Learners

2015-20 Loïs Vanhée(UU)
      Using Culture and Values to Support Flexible Coordination

2015-21 Sibren Fetter (OUN)
      Using Peer-Support to Expand and Stabilize Online Learning

015-22 Zhemin Zhu(UT)
      Co-occurrence Rate Networks

2015-23 Luit Gazendam (VU)
      Cataloguer Support in Cultural Heritage

2015-24 Richard Berendsen (UVA)
      Finding People, Papers, and Posts: Vertical Search Algorithms and Evaluation

2015-25 Steven Woudenberg (UU)

Bayesian Tools for Early Disease Detection

2015-26 Alexander Hogenboom (EUR)
Sentiment Analysis of Text Guided by Semantics and Structure

2015-27 Sándor Héman (CWI)
Updating compressed colomn stores

2015-28 Janet Bagorogoza(TiU)
KNOWLEDGE MANAGEMENT AND HIGH PERFORMANCE; The Uganda Financial Institutions Model for HPO

2015-29 Hendrik Baier (UM)
Monte-Carlo Tree Search Enhancements for One-Player and Two-Player Domains

2015-30 Kiavash Bahreini(OU)
Real-time Multimodal Emotion Recognition in E-Learning

2015-31 Yakup Koç (TUD)
On the robustness of Power Grids

2015-32 Jerome Gard(UL)
Corporate Venture Management in SMEs

2015-33 Frederik Schadd (TUD)
Ontology Mapping with Auxiliary Resources

2015-34 Victor de Graaf(UT)
Gesocial Recommender Systems

2015-35 Jungxao Xu (TUD)
Affective Body Language of Humanoid Robots: Perception and Effects in Human Robot Interaction

====
2016
====

2016-01 Syed Saiden Abbas (RUN)
Recognition of Shapes by Humans and Machines

2016-02 Michiel Christiaan Meulendijk (UU)
Optimizing medication reviews through decision support: prescribing a better pill to swallow

2016-03 Maya Sappelli (RUN)
Knowledge Work in Context: User Centered Knowledge Worker Support

2016-04 Laurens Rietveld (VU)
Publishing and Consuming Linked Data

2016-05 Evgeny Sherkhonov (UVA)
Expanded Acyclic Queries: Containment and an Application in Explaining Missing Answers

2016-06 Michel Wilson (TUD)
Robust scheduling in an uncertain environment

2016-07 Jeroen de Man (VU)
Measuring and modeling negative emotions for virtual training

2016-08 Matje van de Camp (TiU)
A Link to the Past: Constructing Historical Social Networks from Unstructured Data

2016-09 Archana Nottamkandath (VU)
Trusting Crowdsourced Information on Cultural Artefacts

2016-10 George Karafotias (VUA)

Parameter Control for Evolutionary Algorithms

2016-11 Anne Schuth (UVA)
Search Engines that Learn from Their Users

2016-12 Max Knobbout (UU)
Logics for Modelling and Verifying Normative Multi-Agent Systems

2016-13 Nana Baah Gyan (VU)
The Web, Speech Technologies and Rural Development in West Africa - An ICT4D Approach

2016-14 Ravi Khadka (UU)
Revisiting Legacy Software System Modernization

2016-15 Steffen Michels (RUN)
Hybrid Probabilistic Logics - Theoretical Aspects, Algorithms and Experiments

2016-16 Guangliang Li (UVA)
Socially Intelligent Autonomous Agents that Learn from Human Reward

2016-17 Berend Weel (VU)
Towards Embodied Evolution of Robot Organisms

2016-18 Albert Meroño Peñuela
Refining Statistical Data on the Web

2016-19 Julia Efremova (TUE)
Mining Social Structures from Genealogical Data

2016-20 Daan Odijk (UVA)
Context & Semantics in News & Web Search

2016-21 Alejandro Moreno Célleri (UT)
From Traditional to Interactive Playspaces: Automatic Analysis of Player Behavior in the Interactive Tag Playground

2016-22 Grace Lewis (VU)
Software Architecture Strategies for Cyber-Foraging Systems

2016-23 Fei Cai (UVA)
Query Auto Completion in Information Retrieval

2016-24 Brend Wanders (UT)
Repurposing and Probabilistic Integration of Data; An Iterative and data model independent approach

2016-25 Julia Kiseleva (TU/e)
Using Contextual Information to Understand Searching and Browsing Behavior

2016-26 Dilhan Thilakarathne (VU)
In or Out of Control:
Exploring Computational Models to Study the Role of Human Awareness and Control in Behavioural Choices, with Applications in Aviation and Energy Management Domains

2016-27 Wen Li (TUD)
Understanding Geo-spatial Information on Social Media

2016-28 Mingxin Zhang (TUD)
Large-scale Agent-based Social Simulation - A study on epidemic prediction and control

2016-29 Nicolas Höning (TUD)
Peak reduction in decentralised electricity systems -Markets and prices for flexible planning

2016-30 Ruud Mattheij (UT)
The Eyes Have It

2016-31 Mohammad Khelghati (UT)
Deep web content monitoring

2016-32 Eelco Vriezekolk (UT)