Designing for Privacy in a Multi-Agent World

Eric Yu[1] and Luiz Marcio Cysneiros[2]
[1]Faculty of Information Studies
University of Toronto
yu@fis.utoronto.ca
[2] Department of Mathematics and Statistics
York University
cysneiro@math.yorku.ca

Abstract: In a multi-agent world, privacy may have different meaning and significance for different agents. From a system design viewpoint, a practical approach to privacy should allow for a variety of perceptions and perspectives on privacy. Furthermore, privacy must be considered together with all the other requirements - functionality, usability, performance, costs, security, and so on. While there is a growing body of knowledge about privacy issues and how to address them through technical and non-technical means, systematic frameworks are needed to assist system analysts and designers in identifying, analyzing, and addressing these issues. In a networked, multi-agent environment, privacy concerns arise in the context of complex relationships among many human and automated agents. Each agent could have different viewpoints on what notions of privacy apply, and what mechanisms are appropriate for providing adequate privacy, in light of other competing or synergistic requirements. In this paper, we show how the *i** framework can be used to model and reason about privacy requirements and solutions. Agents have privacy goals which are refined, then operationalized into implementable mechanisms, often through dependencies on other agents. To support early-stage design decisions, the impact of alternative solutions are assessed by propagating qualitative evaluations through a dependency network. A example in the health care domain is used to illustrate.

1. Introduction

In today's networked information systems, privacy is of increasing concern. There is a growing body of knowledge about privacy issues, and a range of approaches and technologies for addressing them. From a practical viewpoint, there is no single conception of privacy that can be universally applied. Different organizations, groups, or individuals, under different circumstances and contexts, may well have different perceptions and interpretations of what privacy is, and may settle on different approaches and mechanisms for meeting their privacy needs. In particular, privacy is often traded off against other needs and desires, such as security, cost, convenience, etc. For example, one may be willing to disclose some personal information in exchange for discounts at a reputable retail store.

Privacy issues typically arise within a context of social relationships. In a multi-agent systems framework, an agent makes decisions about the adequacy of its privacy arrangements in relation to the overall relationships it has with other agents.

Consider a multi-agent system for assisting patients requiring long-term care. Software agents representing patients, physicians, hospitals, laboratories, etc. collaborate to achieve effective, high-quality care. Patients do not want their medical records to be seen by unauthorized third parties, especially health insurance companies. But the need to assure privacy may lead to design decisions such as the use of cryptography that can compromise system performance, which might be considered critical by physicians and patients. Various authentication mechanisms may affect usability in different ways, for different kinds of users.

In systems design, privacy, security, performance, maintainability, usability, etc., are usually viewed as non-functional requirements. Non-functional requirements (NFRs) frequently take a backseat to functional requirements. As a result, it is not uncommon for NFRs to be addressed as afterthoughts, and failing to be adequately met in the final product. Functional requirements specify what functions and features a system has to provide, whereas non-functional requirements concern how well the functions are accomplished, e.g., a good response time (performance), how reliable is the software (reliability), or how safe it is to use the system (safety). Errors due to omission of, or inadequately addressed NFRs are among the most expensive and most difficult types of errors to correct [7] [5] [4].

Software development methods for building multi-agent software systems need to provide ways for systematically representing and reasoning about privacy and other requirements as perceived by each agent. The methods should allow for different interpretations of privacy, leading to different privacy mechanisms. As privacy requirements are refined and elaborated, their conflicts or synergies with other requirements need to be recognized. Alternative solutions should be evaluated on how each of them contributes positively or negatively to each of the requirements. This way, decisions can be arrived at in a more systematic way. A systematic approach also encourages the reuse of requirements and design knowledge, and facilitates maintenance and evolution.

The ***i**** framework models relationships amongst social actors from a strategic perspective. Actors depend on each other forming a network of intentional dependencies. In examining this network, one can reason about opportunities and vulnerabilities.

In this paper we will show how ***i**** supports modelling and reasoning about non-functional requirements such as privacy and security in a multi-agent context. For example, in developing a health care information system, one would like to be able to express that a patient depends on the physician for having his expectations regarding privacy to be met. From that starting point, one can systematically analyze in what ways and senses patients depend on physicians, and which possible alternatives the physician has for meeting the patient's expectations. With these alternatives explicitly modeled, one can then assess whether they sufficiently meet patient's expectations of privacy, and if not, be guided to search for further alternatives.

In ***i*,*** privacy, security, and other NFRs are modeled as *softgoals* to be *satisficed* from the viewpoint of each stakeholder. The softgoal concept is used to model quality attributes for which there are no *a priori* clear-cut criteria for satisfaction. Social actors judge whether the attribute is sufficiently met ("satisficed") on a case-by-case basis. Using this approach**,** one can start reasoning about privacy issues starting from the earliest stages of software development.

To illustrate the use of ***i**** to deal with privacy issues, we use an example from the health care domain. Most information systems in use today in health care are provider-centered, i.e., they are developed from the perspective of the health care provider (hospital, physician's clinic, etc.). Many problems can be traced to this provider-centered orientation. For example, one important consequence is the difficulty for patients to take their medical records from one facility to another. A number of initiatives are exploring a patient-centered approach to managing health care information. Some of these assume the use of multi-agent software systems [9], [6]. Clearly, privacy is one of the central concerns to address in the design of these systems.

We will show how one could use ***i**** to model the different privacy concerns that can arise, together with the functional aspects of the software as well as other non-functional aspects like, performance, portability, etc. We will also show how to analyze different alternatives, and which of them would better suit to solve the problem being addressed.

2. Achieving Privacy and Security During System Design

There is a growing body of literature presenting practices, techniques and technologies that can be used to implement and enforce privacy and security in networked environments. However, it is not sufficient for system developers to have knowledge about individual mechanisms. The mechanisms are highly interrelated and furthermore, they interact with other aspects such as usability, performance and availability. When we introduce one of these mechanisms in the software design we may be creating conflicts with other NFRS. These conflicts have to be analyzed and resolved. The literature presents different categorizations for these mechanisms. We here adopt the categorization used by the Organization for Economic Co-operation and Development [8]. Its categorization will guide us when decomposing the concept of privacy. They divide privacy into seven different categories:

- Minimizing the disclosure and collection of personal data;
- Informing users about online privacy policies;
- Providing users with options for personal data disclosure and use;
- Providing access to personal data;
- Protecting privacy through transborder data flow contracts;
- Enforcing privacy principles; or
- Educating users and the private sector.
- Authenticating access

Each of the above categories can be used as a first step in the process of decomposing privacy. To achieve privacy in a system we start by reasoning about the higher levels of privacy that we expect to be satisficed. Once we have chosen which ones will have to be addressed, we have to specialize these goals by refining them into more concrete goals. Different mechanisms will eventually have to be used to "operationalize" the high-level goals for privacy. However, many of these mechanisms when introduced in the software design may conflict with other requirements.

To illustrate this idea we present below some of these mechanisms together with some examples of possible conflicts.

Privacy

Some well-known mechanisms for enforcing privacy [8] [1] include:

- Management of cookies – allowing individuals to limit or prevent the creation of cookies preserves the anonymity.
- Blocking the transfer and collection of automatically generated data – It can be achieved by using anonymous e-mailers, such as Hotmail or Freedom Remailer, or by using an anonymising intermediary such as Anonymizer
- Anonymous Payment Systems – To assure that payments can be done preserving as possible one's privacy one might use payment mechanisms that assure privacy such as Ecash or Mondex.
- Digital Certificates – Issued by a trusted source uses public key cryptography techniques to establish personal attributes with little or no disclosure of the party's true name or any other identification.
- Anonymous Profile – Can be used when websites are only interested in collecting profiles. Doubleclick and Clickstream are examples of the use of such mechanism.
- Posted Privacy Policies – Aims to let the user knows what are the privacy policies of a particular website. Changes in the policy might be communicated by either email notification or porting the changes in the website
- Choice of Data / Consent – Allows the user to choose what data should be collected or not.

Choosing among the different mechanism is not an easy job. If one is designing for privacy he must be able to reason about the several alternatives of mechanisms to use. For example, will the use of management of cookies be enough for the level of privacy one is seeking? If the software is supposed to run mostly in intranets maybe the trust users have in the company together with the management of cookies be enough. If that is so, what is needed to implement that design decision? We may probably need the browser to be configured automatically. And what if we have chosen to adopt some type of consent mechanism allowing the user to configure on-line what data he wants or not to be disclosed? What design implications that would lead?

Also, some conflicts may arise when we decide to use some of the above mechanisms. For example using anonymous payment system may conflict with performance aspects if this mechanism turns out to be accept by a critical mass of merchants. Security is also a concern since anonymous payment may facilitate money laundering.

In the case of digital certificates for example, its use may conflict with maintainability requirements since attributes might change over time and keeping them accurate may be a challenge.

Being able of modeling all the different alternatives and their contributions (both positive and negative) towards achieving privacy would lead us to more substantiated design decision.

Security

The same way we did with privacy, security will be decomposed in high level goals that might contribute, individually or in group, to satisfice security requirements. Most of them were extracted from [3]

- Access
- Confidentiality
- Integrity
- Availability

For operationalizing the above security goals, many mechanisms can be used (e.g. [18],[20],[19]).

- ❖ Public Key Infrastructure (PKI) – Specialized agents can be used as key certification authorities. These agents are able to handle multiple certificate formats and trust hierarchies leading to an interoperability of agent systems using multiple PKI.
- ❖ Security-Pass style – Using this mechanism one can efficiently represent the security context for any method activation
- ❖ SSL Protocol underneath the agent communication layer – Aims to keep details related to communication security transparent from the application and to facilitate the use of off-the-shelf trustworthy technology.
- ❖ Assign PKI to agents – Using PKI we can make agents uniquely identifiable and thus allow agent to be sure about who the are talking to.
- ❖ Integrity mechanism – For example by using message signatures to ensure the integrity of a message.
- ❖ Authentication Mechanisms – Such as Static passwords, Dynamic Passwords and Biometrics.

Let us here take for example the use of authentication mechanisms. Biometrics for example may assure a better authentication but might hurt the requirements for cost and availability since they are still expensive and not as compatible with many devices as one might want it to be.

In the case of integrity mechanisms we may eventually face a conflict with performance requirements since the introduction of signatures in the messages may lead to an overhead that could not be acceptable

3. Modelling Privacy and Security in i*

Let us take for example the relationship between patient and physician. We use the exemplar proposed for agent-oriented software development methodologies [15] that is based on the Guardian Angel Project [10]. The Guardian Angel project aims to "construct information systems centered on the individual patient instead of the provider, in which a set of "guardian angel" software agents integrates all health-related concerns, including medically-relevant legal and financial information, about an individual. This personal system will help track, manage, and interpret the subject's health history, and offer advice to both patient and provider. Minimally, the system will maintain comprehensive, cumulative, correct, and coherent medical records, accessible in a timely manner as the subject moves through life, work assignments, and health care providers.

Patients expect to be assessed by physicians and to have privacy regarding all the information provided to physicians along with any medical information the physician might collect or produce. Figure 1 shows the SD model representing that. The SD model depicts a process as a network of dependency relationships among actors. In ***i****, a dependency is a relationship in which one actor (the *depender*) depends on another actor (the *dependee*) for something (the *dependum*) to be achieved. A dependum can be a goal, task, resource, or softgoal, reflecting the types of freedom allowed by the relationship. A goal dependency is one in which one actor depends on another to bring about a certain condition or state in the world, while the depended actor (the dependee) is free to, and is expected to, make whatever decisions are necessary to achieve the goal. Thus, it also indicates that one actor does not care how the other actor will achieve this goal. In Figure 1 we can see that the actor Patient depends on the actor Physician to have the goal Be Assessed achieved and also to have the softgoal of Privacy regarding medical records to be accomplished.

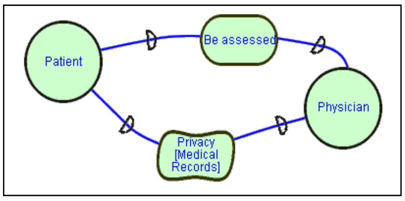

Figure 1 – Privacy between Patient and Physician

While the SD models focus on external relationships among actors, the SR models describe the intentional relationships that are "internal" to actors, in terms of process elements and the rationale behind them. The generic notion of actor may be differentiated into agents, roles, and positions. Rationales are modelled through means-ends relationships, task decompositions, and softgoal contributions [13].

Managing patient's record is part of the physician's job when assessing the patient. Using an SR model we can detail how this management can occur and how the expected privacy will affect it, i.e. which efforts the physician might undertake to satisfice the softgoal dependency that the patient has on him. Figure 2 illustrates the reasoning. Management of patient's record can be done in two different ways, either manually or using electronic records, i.e. software systems. Many physicians may decided to keep managing patient's record manually as they do today because they do not trust in software systems to handle such a delicate thing as the patient's record. Others may be confident enough to adopt electronic records or might even be compelled to use it either by their bosses or eventually by law.

When doing it manually there is a task for assuring confidentiality that is considered to help Privacy aspects. It helps because patients trust in their physicians so if privacy depends only in the physicians' efforts for keeping the records private it may be enough for the patient.

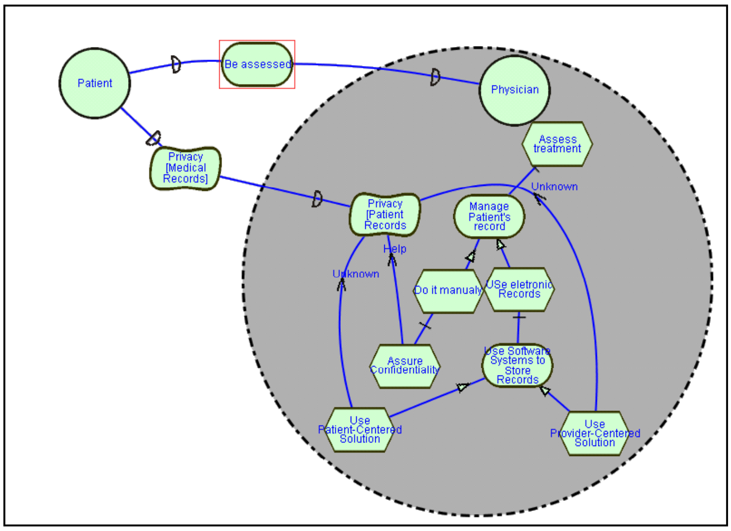

Figure 2 – Starting to Modelling Privacy

On the other hand, when using software systems we might have two different options. First we can adopt a patient-centered solution where all the patient's record will be in patient's hand. The second alternative would be to use today's solution, provider-centered solution, of having it controlled by health care providers. Initially the contribution of each of these solutions to the privacy softgoal is considered *unknown* since there are no insights about what each alternative would represent.

Since we have two alternatives for managing the patient's record we have to model these alternatives, first using basically a SD model to have a broader view of the problem and later refining it into SR models. Figure 3 shows the broader model. We got to that model reasoning about which other actor would have to be involved to address each possible solution. For the patient-centered solution we introduced the Guardian Angel software agent. For the provider-centered solution we initially introduced the Hospital Software System agent. Later, we realized that many hospitals would use software companies not only to provide the software but also to administrate it, enabling the hospital to concentrate in its area of expertise. For representing that, we introduced a new type of dependency, the resource dependency. This type of dependency means that an actor depends on another to have a resource delivered (either physical or informational). In this case for example, the physician will depend on software to have the medical records delivered.

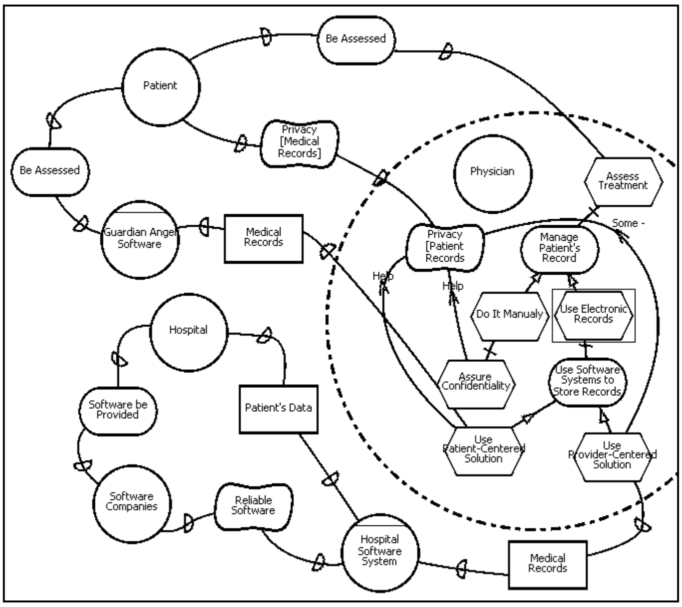

Figure 3 – Exploring Alternatives

At this point we may change the contributions of each alternative to Privacy. The patient-centered solution would *help* to achieve Privacy since having the data controlled by the patient leave little room for Privacy problems, at least at a first glance. On the other hand, a provider-centered solution would have a negative impact on Privacy (*some* -) since patients may not completely trust in the hospitals and even less in third-party companies.

As we are modeling different actors, we can model different viewpoints for Privacy. For example, we can represent Privacy through the viewpoint of web sites together with Privacy through the viewpoint of customers that use these web sites. Privacy may be differently decomposed to each actor's viewpoint and thus different operationalizations for Privacy can be found representing each actor's viewpoint. Necessary tradeoffs to satisfice both viewpoints can be made targeting a solution that satisfices the stakeholders the better way possible.

We must then investigate how Privacy would be translated to each of the software agents involved. We will in fact represent three different viewpoints for Privacy, the patient viewpoint, the hospital/software provider viewpoint and the Physician viewpoint. Physicians' viewpoint for privacy will basically be restricted to his own efforts to assure patients' Privacy. Actually, as we are adopting electronic records, satisficing Privacy will be totally dependable on satisficing Privacy within the chosen approach. We will detail the existing model into SR models as depicted in Figure 4. We will first introduce the Privacy softgoal in each of the software agents.

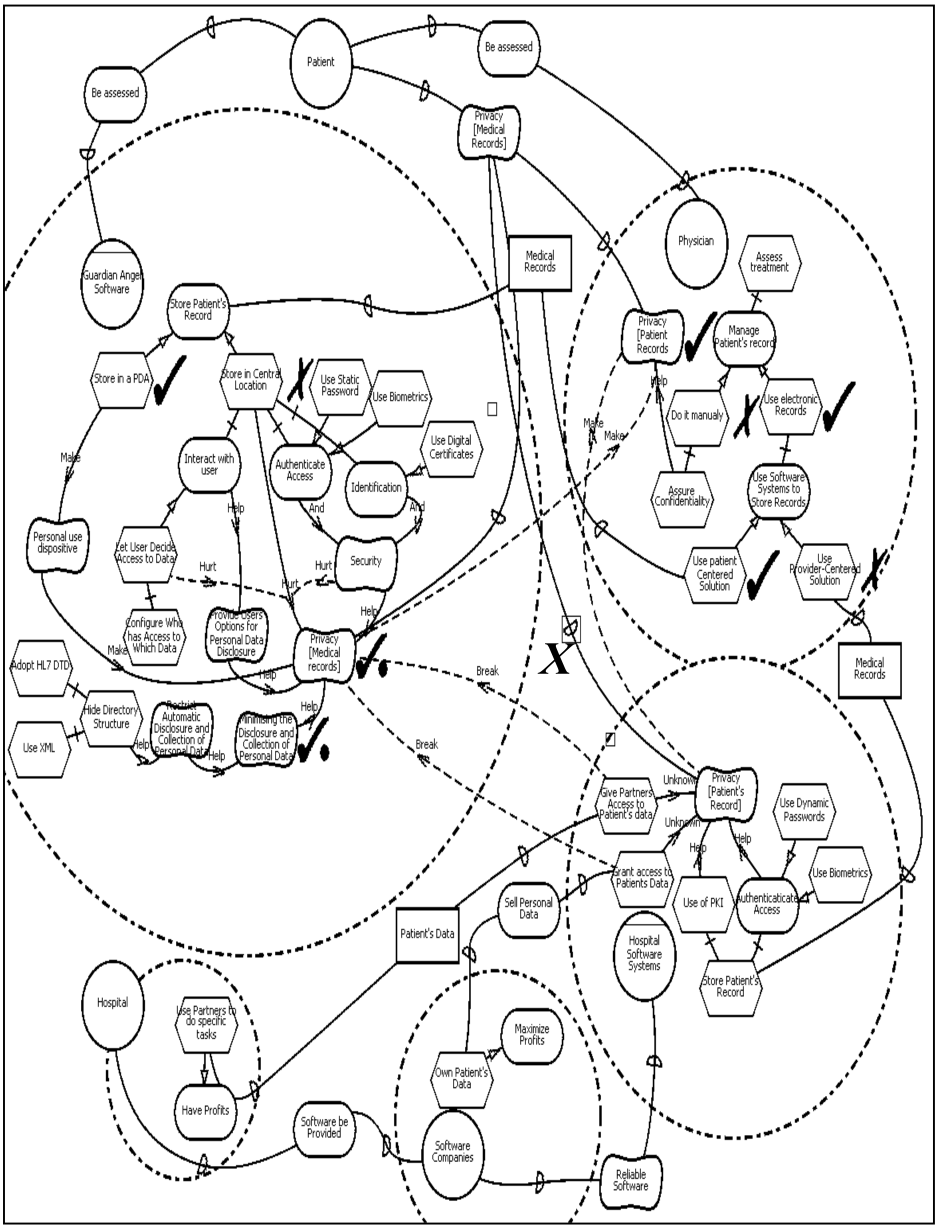

Figure 4 – Privacy Examined in Detail

The Privacy softgoal that patient has on physician will now also depend on the Privacy softgoals on the Guardian Angel Software and on the Hospital software. We have then to further decompose the Privacy softgoal. . A first attempt to that must start using the general categories for addressing Privacy depicted n Section 2. Further decomposition may use specific mechanisms that may fit to the category(ies) used. As no clear pattern was found for that we live it open for individual choice. We can see in Figure 4 that in this case we chose to decompose first Privacy into *Providing Users Options for Personal Data Disclosure*, *Minimizing Disclosure and Collections of Personal data.*

In the Guardian Angel software we may have two different alternatives for storing the patient's record. We may either store it in the PDA that will be used by the patient or store it in a central location. We can see (Figure 4) that the latter would contribute negatively to Privacy (*hurt*) since it may be more vulnerable to external attacks. In the other hand, storing it in the PDA (personal digital assistant) would *make* the softgoal. Being a personal device, the PDA is most likely to be used only by the patient or by people he might trust and to whom and concerns about Privacy could be ignored. It is true that when eventually remotely connected, supposing these capabilities are offered, the PDA may be vulnerable to external attacks, but in this case other considerations than Privacy would have to be made and it is out of our focus at the moment. One might suggest that storing patient's record in a central location would allow the use of more powerful hardware leaving room for patients to be able to choose who should have access to what information. This would contribute to privacy and it is modeled as a *hurt* correlation link (dotted line) to the *hurt* contribution from storing in a central location to privacy.

By introducing security mechanisms we can also diminish the vulnerability of storing in a central location. In this particular case we decided to address security by using Authenticate access and Identification. Refining the *authenticate access* goal we can think about two different approaches, the use of static passwords or the use of biometrics. Identification goal would be refined into the use of digital certificates to ensure that the software is being accessed by the correct person. Security would contribute positively towards Privacy and it is shown with the *hurt* correlation link from Security to the *hurt* contribution from storing in a central location to Privacy softgoal.

On the other hand, we can see that to operationalize the Privacy softgoal; i.e. further decomposing the sub-goals that first decompose Privacy, we want to have mechanisms like choice of data, authentication and not disclosing personal data. Refining the latter leads us to keep every directory structure hidden since the simple fact of having a subdirectory with a name of a disease might *hurt* patient's Privacy. Imagine for example a patient navigating his records and that someone is looking. Imagine now that at the end of the directory appearing in the browser you have HIV. The simple fact that someone knows you have taken the HIV test can be enough to get you in trouble. To deal with that the Guardian Angel Project proposes to combine the use of XML and HL7 DTD [17].

Looking through the hospital and software provider viewpoints things are not so simple. The software agent would use PKI and authentication to achieve the Privacy softgoal. However, Hospitals want to use partners like clinical laboratories or image diagnosis laboratories to do part of the patient's assessment. To do that, access should be granted to patient's records. That would compromise Privacy in the patient's viewpoint since it cannot be granted that hospital partners will not use patient's data in such a way that would be against patients will. Regarding the Privacy through the hospital software system's viewpoint granting access to partner would have a neutral impact (represented as unknown).

The software companies' viewpoint can be even worst. In order to maximize profits they may want to own patient's data so they can sell them the way they want. Again, through the hospital software system's viewpoint it would have no clear impact on privacy. On the other hand it would definitely compromise the Privacy in patient's viewpoint. Therefore, at this point we decide to go with the patient-centered alternative storing it in the PDA. This is represented by the symbol of denied (*X)* next to the dependency link from Patient's Privacy softgoal to Hospital Software Systems Privacy softgoal denoting that this dependency will no be enforced. As dependencies are by default satisfied, the absence of any symbol in the other dependencies means that these dependencies will be satisfied. It remains to decide whether we use the PDA to store the data or a central location.

Although the use of Privacy mechanism can improve Privacy when storing in a central location, Privacy can be more extensively granted if we store data in the PDA. By using ***i**** modeling we could represent up to now the different alternatives we have and how they would contribute positively or negatively to Privacy. Up to this point we can see that the patient-centered alternative storing data in a PDA is the alternative that presents the best contribution and therefore should be chosen to be implemented. The presence of the denied symbol next to the task denoting the central location storage means that this alternative will not be adopted. On the other hand, the existence of the satisficed symbol (√) next to the store in a PDA task denotes that this alternative of design has been chosen to be implemented.

Of course Privacy and Security are not the only concern in a complex project like that. Many others NFRs such as availability, performance and security can play important role in design decisions and should therefore be modeled and analyzed. Although we are presenting privacy and security modeling separately from other NFRs, we do that only for the sake of simplicity. In a real situation privacy would be modeled together with other NFRs

4. Privacy and Other NFRs - Reasoning among different alternatives

Aside from Privacy we have also to consider others NFRs that might directly impact the design and may call for tradeoffs to be made. In the example we are using, Availability is one major concern from physicians' viewpoint. If on the one hand having patient's record stored in the PDA up to now is the best alternative to be taken, on the other hand, if we consider the need for patient's record to be available we may have to carry out further investigations. Availability of medical records is important to allow physicians to have quality in their assessment (a softgoal that decomposes the assess treatment task). In a normal situation, the patient would be able to provide the physician all necessary data by allowing the PDA to interface with the physician's computer and the latter to retrieve the necessary information as long as the information solicited have been authorized by the patient to be transferred. This alternative would still be compliant with the decision of having patient's record stored in the PDA. However, during an emergency, the patient may not have the physical ability to give access to his PDA or may even be unconscious. Figure 5 portrays this as a *hurt* contribution link from the task store in a PDA within guardian angel software to the Availability softgoal within physician together with a belief stating this knowledge that *helps* in the *hurt* contribution.

On the other hand, Storing in a central location will contribute towards availability because data could be accessed by accredited hospitals and authorized physicians. Since Availability is considered in this case to be as important as Privacy, the final decision was to store on both the patient's PDA and in a Central Location.

Another NFR requiring tradeoffs is Usability. Reasoning about it, we realize that some tasks that help the Privacy softgoal would hurt Usability. Configuring access may be a challenge for many patients while to authenticate access depends on what alternative of authentication we decide to use. The use of password authentication may *hurt* usability because remembering and entering passwords long enough to be secure may be difficult to many patients. In the other hand, the use of biometrics would *help* usability because they do not call for any effort from the patient to

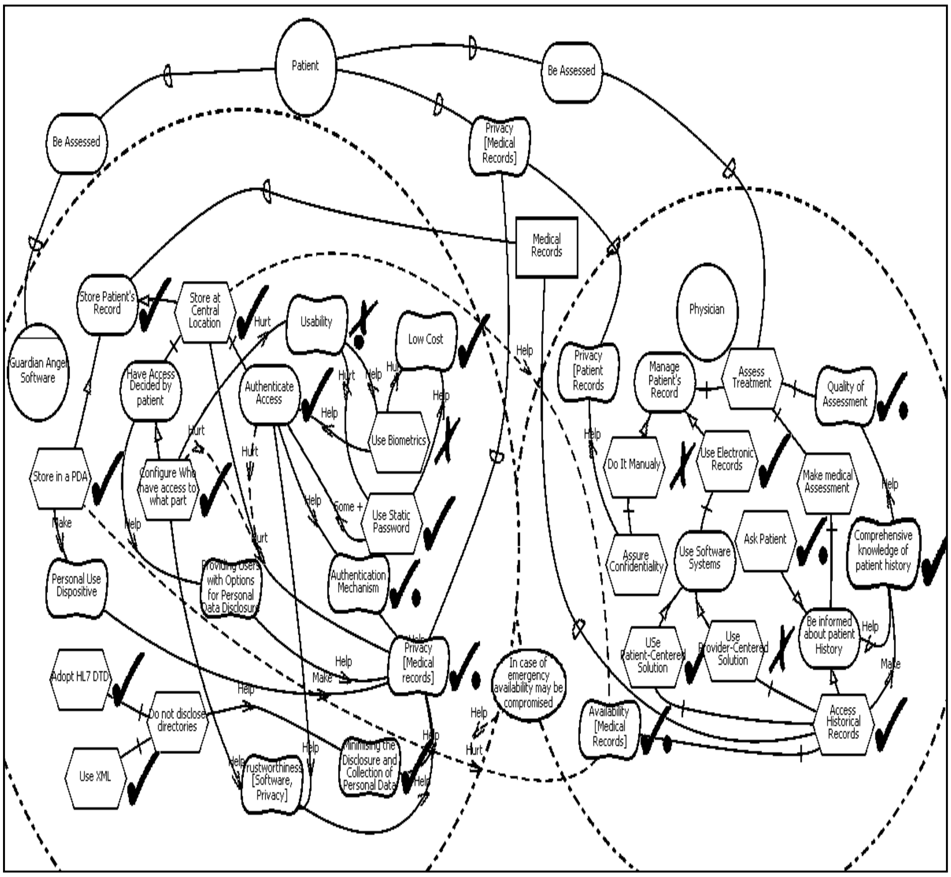
Patient
Be Assessed
Be Assessed
Privacy [Medical Records]
Medical Records
Store Patient's Record
Store at Central Location
Hurt
Usability
Low Cost
Guardian Angel Software
Have Access Decided by patient
Authenticate Access
Hurt
Help
Hurt
Help
Help
Use Biometrics
Hurt
Hurt
Store in a PDA
Configure Who have access to what part
Help
Some +
Use Static Password
Make
Help
Hurt
Personal Use Dispositive
Providing Users with Options for Personal Data Disclosure
Authentication Mechanism
Help
Make
Privacy [Medical records]
Adopt HL7 DTD
Help
Do not disclose directories
Help
Use XML
Help
Trustworthiness [Software, Privacy]
Minimising the Disclosure and Collection of Personal Data
Help
Help
In case of emergency availability may be compromised
Help
Hurt
Help
Physician
Privacy [Patient Records]
Manage Patient's Record
Assess Treatment
Quality of Assessment
Help
Do It Manualy
Use Electronic Records
Make medical Assessment
Help
Assure Confidentiality
Use Software Systems
Ask Patient
Comprehensive knowledge of patient history
USe Patient-Centered Solution
Use Provider-Centered Solution
Be informed about patient History
Help
Make
Availability [Medical Records]
Access Historical Records

Figure 5 – SR Model After Represent and Reasoning About Alternatives

use it. However, it may *hurt* another softgoal that represents the need to keep the Costs low.

Of course, these NFRs would not be the only ones involved in a system like this. Performance for example could be impacted if some sort of cryptography is used as well as Security has to be a concern since if we decide to store it in a central location like a web site, additional Security measures might have to be taken. However, for the sake of simplicity we will restrict the example to the NFRs modeled in Figure 5.

Figure 5 portrays not only the different alternatives but also the design decisions taken. For example, we see when we finally chose to satisfice the storage of patient's data in the guardian software we decided for doing that using a central location to store it and also store in a PDA. Adopting the authenticated access and allowing the patient to configure who would have access to what data, contribute to diminish the negative impact of this decision. These two mechanisms will also contribute to Trustworthiness of patient regarding how the software can assure Privacy, which in turn will *help* to satisfice the Privacy softgoal. It is true that on doing so we will be *hurting* Usability concerns but Privacy was considered, in this case, to be more relevant than Usability. Finally, Usability is considered to be less important than keeping Costs low and thus we keep the option of using password authentication instead of adopting biometrics solutions.

We have only tackled part of the problems that might arise when reasoning with Privacy. When carrying out a comprehensive reasoning, several other aspects may be tackled. For example we might have to deal with particular kind of patients that because of the nature of their disease might need to deal with different instantiations of the problem. They might eventually demands a more careful approach for Usability, demanding different alternatives for privacy to be searched or different considerations on the costs. Another possible example typical to the health care domain comprises Security. Many applications demand a log of access to be kept registering not only which functionalities the user has accessed but also all the data that was modified so if any problems arises in the future one can trace back the data input and find the responsible for the problem. That would conflict with Privacy concerns and would have to be taken to the stakeholders to analyze the possible tradeoffs.

5. Conclusion

This work argues for the need for systematic design frameworks for modelling and reasoning about privacy, security and other NFRs. We showed examples using the ***i**** framework to illustrate how one can model privacy as softgoals in order to assess the different alternatives to satisfice each notion of privacy and how each alternative would contribute positively or negatively for achieving privacy.

The ***i**** framework also allows one to explore different levels of abstraction by using SD and SR models, easily moving from one level of abstraction to another. Tracing the impacts of one change is also improved through the use of the ***i**** framework since we can simply represent one alternative previously satisficed as denied and vice-versa and thus evaluate the impact of these decisions on the design.

The ***i**** framework is complementary to other approaches addressing privacy. In [16] is shown some of the challenges of addressing privacy for agent-based e-commerce software systems together with a policy-driven approach for privacy negotiation. In [1] a taxonomy of privacy for web sites is shown with some high level categorizations together with many goals that at some level can help the designer on choosing among different alternatives for each case. In [8] is presented an inventory of instruments and mechanisms to address privacy on global networks. All the above works are important to bring to light the different approaches one might have for addressing privacy. However, having a comprehensive list of mechanisms without being able to understand their impact in the whole software design can frustrate the efforts for good quality systems. In this paper we have shown how to use ***i**** as a basis for modeling and reasoning about privacy. Works, like those mentioned above, are used to categorize privacy in such a way that it ca help us on decomposing privacy into high-level sub-goals that can lead to privacy satisficing. The ***i**** framework can be used to a preliminary analysis of the domain and its inherent social relationship being later detailed with the many well know mechanisms to ensure privacy. The models can be used to express the different mechanisms one might consider to satisfice privacy within a domain and represent all the consequences of each alternative. The ***i**** approach facilitates and encourages to do so.

We have shown an example from the health care domain showing how different alternatives can be modeled to satisfice privacy and how they would contribute not only to privacy but also to security and other requirements as usability, availability and cost. We have also shown that some alternatives might even contribute to privacy satisficing indirectly, e.g. by enhancing the trust the patient would have on the software. Also, different perspectives for the same problem can be modeled as we showed here by focusing on the different viewpoints patients and hospitals might have. This is particularly important for the web domain because web providers' viewpoint may not match customers' viewpoints. Having modeled the different alternatives and their impacts one can go through a more detailed analysis of the domain and make design decisions in a less intuitive way.

As ***i**** allows softgoals and their operationalization to be organized in form of knowledge base catalogues, previous experience can be reused in the future.

The framework has been used in many different domains as telecommunication, smart cards and health care domains, including real-life case studies. However, practical use of ***i**** for dealing with privacy is still an issue.

Future work includes studying more deeply the interrelationship between privacy and trust and to improve the existing prototype tool that support the modeling and reasoning based on ***i****.

References

[1] Antón, A.I. and Earp., J.B. " A taxonomy for Web Site Privacy Requirements" NCSU Technical Report TR-2001-14, 18 December 2001.

[2] Barber, K.S. and Kim, J. " Belief Revision Process Based on trust: Agents Evaluating Reputation of Information Sources" in Proc. Autonomous Agents' , Workshop on Deception, Fraud and Trust in Agent Societies, 2000 Barcelona.

[3] Chung, L., Nixon, B., Yu, E. and Mylopoulos,J. “*Non-Functional Requirements in Software Engineering”* Kluwer Academic Publishers 2000.

[4] Cysneiros,L.M., Leite, J.C.S.P. and Neto, J.S.M. *“A Framework for Integrating Non-Functional Requirements into Conceptual Models*” Requirements Engineering Journal – Vol 6 , Issue 2 Apr. 2001, pp:97-115.

[5] Ebert, C. “*Dealing with Nonfunctional in Large Software System”s*. Annals of Software Engineering, 3, 1997, pp. 367-395.

[6] Lanzola, G., Gatti,L., Falasconi, S., Stefanelli, M. “A Framework for Building Cooperative Software Agents in Medical Applications.” Artificial Intelligence in Medicine 16 (1999) pp:223-249

[7] Mylopoulos,J. Chung, L., Yu, E. and Nixon, B*., “Representing and Using Non-functional Requirements: A Process-Oriented Approach”,* IEEE Trans. on Software Eng, 18(6), pp:483-497, June 1992

[8] “Inventory of instruments and mechanisms contributing to the implementation and enforcement of the OCDE privacy guidelines on global networks” Head of Publications Services, OECD, 2 rue-André-Pascal, 75775 Paris Cedex 16, France.

[9] Riva, A. et all “The Personal Internetworked Notary and Guardian” Internation Journal of Medical Informatics 62 (2001) pp:27-40.

[10] Szolovits, P., Doyle, J., Long, W.J. “Guardian Angel:Patient-Centered Health Information Systems” Technical Report MIT/LCS/TR-604, http://www.ga.org/ga/manifesto/GAtr.html

[11] Wallach, D.S., Appel, A.W. and Felten, E.W. “ SAFKASI: A Security Mechanism for Language-Based Systems” *ACM Transactions on Software Engineering and Methodology*, volume 9, number 4, October 2000.

[12] Wong,H.C. and Sycara, K. “Adding Security and Trust to Multi-Agent Systems” in Proc. Autonomous Agents’ 99, Workshop on Deception, Fraud and Trust in Agent Societies, 1999 Seattle pp:149-162

[13] Yu, E. “Agent-Oriented Modelling: Software Versus the World” Agent-Oriented Software Engineering AOSE-2001 Workshop Proceedings. LNCS 2222.

[14] Yu,E. and Liu, L. “Modelling Trust for System Design Using the *i** Strategic Actors Framework” In: Trust in Cyber-Societies - Integrating the Human and Artificial Perspectives. R. Falcone, M. Singh, Y.H. Tan, eds. LNAI-2246. Springer, 2001. pp.175-194.

[15] Yu,E., Cysneiros.L.M., “Agent-Oriented Methodologies-Towards a Challenge Exemplar” in Proc of the 4th Intl. Bi-Conference Workshop on Agent-Oriented Information Systems (AOIS 2002) Toronto May 2002.

[16] Korba, L. “Privacy in Distributed Electronic Commerce” in Proc. of the 35th Hawaii Int. Conf. on System Science, Jan, 2002

[17] HL7 SGML/XML Special Interest Group. http://www.mcis.duke.edu/standards/HL7/committees/sgml/index.html

[18] Poslad, S. and Calisti,M. “Towards Improved Trust and Security in FPA Agent Platforms” in Proc. Autonomous Agents’ , Workshop on Deception, Fraud and Trust in Agent Societies, 2000 Barcelona pp:87-90

[19] Dan S. Wallach ,Andrew W. Appel,Edward W. Felten *ACM Transactions on Software Engineering and Methodology*, volume 9, number 4, October 2000.

[20] Wong,H.C. and Sycara, K. “Adding Security and Trust to Multi-Agent Systems” in Proc. Autonomous Agents’ 99, Workshop on Deception, Fraud and Trust in Agent Societies, 1999 Seattle pp:149-162