

A Privacy Framework for the Personal Web

Reza Samavi¹, Mariano P. Consens¹, and Thodoros Topaloglou²

¹ MIE, University of Toronto

² Rouge Valley Health System, Toronto, Canada

Abstract. User-centric privacy management is an important component of the Personal Web, and even more so in the context of personal health applications. We describe the motivations behind the development of a personal web privacy framework and outline a layered model for self-management of privacy in the context of Personal Health Record applications. In this paper we provide an overview of our framework. The privacy goals and settings mediator model addresses the understandability problem of privacy agreements and settings by supporting the users' privacy decision-making process. This model provides privacy experts with the tool support to encode their knowledge and fill the gap between the end-users' high-level privacy intentions and what personal health applications offer as privacy features. The second model in our framework, smart privacy model, is an ontological model that supports privacy enforcement. The model provides interoperable and computer interpretable translations of privacy settings, allowing the privacy settings selected by a user, to be translated as enforceable constraints on the data and processes of a personal workflow.

Keywords: privacy model, user privacy preferences, Personal Health Record; goal-oriented modeling, Ontology, Process Specification Language.

1 Introduction

Personal Web is an emerging research topic driven by the transformation of the Internet and web from the way users currently interact with and navigate resources in the web, to a smart paradigm mainly centered on users' experience [1]. The main goal of the Personal Web is to empower users, as individuals, seamlessly and smartly self-manage the vast amount of web resources and services to achieve their personal goals [1]. A user-centric perspective of service utilization requires users to play an active role in the process. The promise of the personal web is to make this shift socially and cognitively viable. Such a perspective brings new challenges into the design and architecture of web applications.

In this paper, we investigate the problem of privacy support in Personal Health Record (PHRs) as an emerging Personal Web application. PHR have been growing toward becoming platforms for an extensible ecosystem of personal health applications. PHRs are open platforms with application-programming interfaces (API). Service providers use these APIs to augment the platform with new applications and services [2]. The main goal of these applications is to empower users to utilize PHR not only for the purpose of storing and retrieving health and life style information, but also as a medium to create personal workflows to accomplish their personal health goals

[3]. Communicating an individual's health data with clinicians, participating in clinical research, or partially sharing health data on social networks [4], are all examples of leveraging PHR platforms for personal goals [5]. With the shift in the PHR's role, users remain loyal to the platform while the third party applications are easily substitutable [2]. The privacy implications of this new PHR architecture are multifold. In this paper we address two aspects of user privacy management in PHR context, the users' privacy configuration problem and the problem of enforcing the configured privacy settings when multiple services over a web platform are utilized.

1.1 Privacy Configuration

For every health-service there are multiple service providers that can become part of the personal workflow. The decision criteria for users to prioritize one application over another are based on the application's cost, value [2], and more importantly the extent that the service respects users' privacy goals [6]. Thus the users' ability to self-manage their privacy and comprehend the consequences of privacy settings in such workflows becomes a core requirement of the Personal Web. However, supports for the self-management of privacy in existing PHRs platforms are primitive and insufficient. When we start to use a service, the only option offered is often to push the "I agree" button at the end of a long legal privacy text (which in most cases is left unread [7]). A number of other applications offer privacy options based on a growing number of privacy features. Nevertheless, these features usually reflect the system's perspective instead of the privacy desires and expectations that a user would want to achieve. Privacy experts can offer users advice to help configure their privacy settings, but there is a lack of tools to support the task of privacy configuration.

As the comprehensibility of privacy agreements and the configuration of privacy settings have become daunting tasks, we propose a solution that seeks the comfort of conceptual modeling techniques. In this paper, we propose the Privacy Goals and Settings Mediator (PGSM) model, a privacy model that helps users to comprehend the privacy settings when employing multiple services over a web platform. The model is based on the i^* multi-agent modelling technique [8]. The i^* modeling technique is originated in the software engineering community, where conceptual modeling is regarded as a tool for engineers and designers to promote a common understanding of a subject matter and facilitate a complex design process [9], [10]. We believe that conceptual models and modelling is equally valuable in order to understand and manage privacy.

The i^* modeling technique is utilized to model the environment and the goals of agents involved in a privacy sensitive interaction, creating a privacy goals and settings mediator model (PGSM). The parties involved in an interaction within a context are represented as the i^* agents who depend on each other to achieve some goals or perform some tasks. Goals are used to express the purpose or utility of an interaction, as well as the qualities associated with a purpose. The users' perception of their privacy are expressed in terms of goal-models of multiple agents. These goal models link the privacy features offered by services to the high-level users goals. The goal-structure allows designers as well as users to reason about how the changes in privacy features affect the users' goals. The achievement or violation of privacy is determined by evaluating the degree of satisfaction of the users' goals.

There is a software tool (OpenOME [11], implemented as an Eclipse plug-in) implementing the *i** framework. The software provides a modeler with the ability to check the model in terms of the satisfaction of each agents goals as circumstances change. We leverage the tool in PGSM to demonstrate how the knowledge of the privacy experts can be encoded into the model as part of the system design process (design-time). We conducted qualitative evaluation of the PGSM model in terms of the model contribution to the comprehensibility of the privacy configuration task performed by PHR users when the application is utilized by the users (run-time). We interviewed privacy experts and we found that they see value in using the PGSM model in order to serve end-users needs. The evaluation results have been reported in [12].

1.2 Privacy Enforcement

When a personal workflow is executing, the configured user's privacy settings, needs to be enforced by multiple services. In other words, the precise constraints on resources or actions that a service provider commits to respect need to be determined over the course of a workflow execution. Classical privacy policy languages for policy enforcement (e.g. P3P[13], XACML[14]) are either suffer from insufficient expressive power, semantic incompatibility, or are too cumbersome to be used in a personal workflow [15]. The second model proposed in this paper addresses the problem of privacy settings enforcement when multiple participants are involved in a personal workflow. We propose the Smart Privacy Model (SPM) in which a modular and extensible ontology provides an unambiguous, interoperable and computer interpretable description of the privacy constraints over the data and processes in a personal workflow.

The Smart Privacy Model maps the output of PGSM to a process ontology that provides the underlying semantics for the enforcement of privacy constraints across multiple services. The PGSM model maps the semantics of the users' high-level privacy concerns and desires to a set of system-level privacy settings. However for substitutable services in a PHR platform to be able to consume these settings at run-time, the settings need to be expressed as sharable and reusable knowledge. In SPM the privacy constraints are expressed declaratively as parameterized first order axioms. We built the model by extending the Process Specification Language (PSL) ontology [16]. As a proof of concept, in this paper, we demonstrate by an example scenario that common privacy constructs (e.g. conditional access, obligations, and norms) can be expressed as constraints on run-time sequences of behaviour execution.

1.3 Research Contributions

The contributions of this research are threefold. First, we identify the gap between the PHR users' privacy goals and the system's privacy features and propose the PGSM model and methodology to fill this gap. Second, the model provides a novel solution for capturing the privacy knowledge of experts and sharing this knowledge. Third, by designing the logical privacy model we were able to translate users' goals and concerns to reusable and interoperable rules and constraints in the operational level of a personal workflow at run-time. This model contributes to the users' privacy management task by

allowing privacy preferences to be expressed once and used by multiple services. Furthermore, from the systems' perspective, privacy constraints in the SPM are expressed using the same semantics constructs available to express all other general process constraints such as task ordering, task concurrency, and task decomposition. This approach allows the design for privacy to be embedded into the design of the application itself as proposed by the principles of *Privacy-by-Design* [17].

We expect with the support that the PGSM and SPM models provide for explicit representation of multiple actors, their goals and desires, and refinement of those goals in an operationalized level as enforceable constraints, would benefit privacy experts to explore and encode PHR privacy-sensitive usage scenarios. Results of the initial survey of privacy experts (Reported in [12]) has been promising, yet more comprehensive study on usability and usefulness of the model for privacy experts would benefit personalized privacy research community.

The remainder of this paper is organized as follows. In Section 2, we describe the architecture of the personal web privacy framework. A motivating scenario is introduced in Section 3. PGSM model and Smart Privacy models are described using the motivating scenario in Section 4 and 5, respectively. Related work is presented in Section 6. We conclude this paper and provide some future directions in Section 7.

2 Personal Web Privacy Architecture

Existing formal privacy policy languages (e.g. P3P [13], XACML[15], EPAL[15]) and privacy logical models (e.g. [18], [19]) are different in terms of their expressive power and scope. However, the key underlying assumption of these languages is that users' privacy goals and concerns are similar to the system privacy rules and constraints. Thus they can be expressed with the same level of abstraction. This is understandable in the classical web realm, where supports for privacy are mainly provided to protect websites and institutions from being liable in case of breaching the privacy laws and regulations rather than addressing users' privacy needs [20].

We identify two problems in the current privacy architecture of the personal web application (such as PHRs). First, as shown in Fig. 1, in the current architecture users are required to configure their privacy in the system context directly. Second, even at the system level, for every single service users need to interact with the different services repeatedly in order to define their privacy settings.

Users' privacy concerns are usually high-level, informal, and negotiable, while the privacy features offered by a system are detailed, strict, and binding. Systems usually do not offer enough support for ensuring that the choices selected by users will achieve the user's intents and desires. Thus, the first design goal of the personal web privacy architecture is to facilitate the users' privacy configuration task in terms of understanding the privacy features and the consequences of sharing personal information.

When the privacy settings for a given service are realized by a user, the user should not be required to reconfigure the privacy settings of her personal workflow if she decides to substitute a service with another service while nothing has changed in terms of her privacy preferences. Therefore, the second design goal of the personal web privacy architecture is eliminating the repeating task of privacy configuration by providing a run-time support for reusing the semantics of the user's privacy settings.

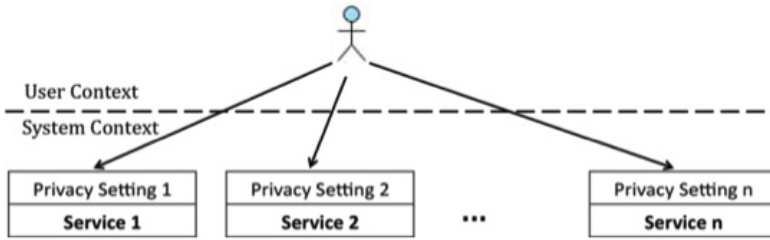


Fig. 1. Current Privacy architecture in PHR

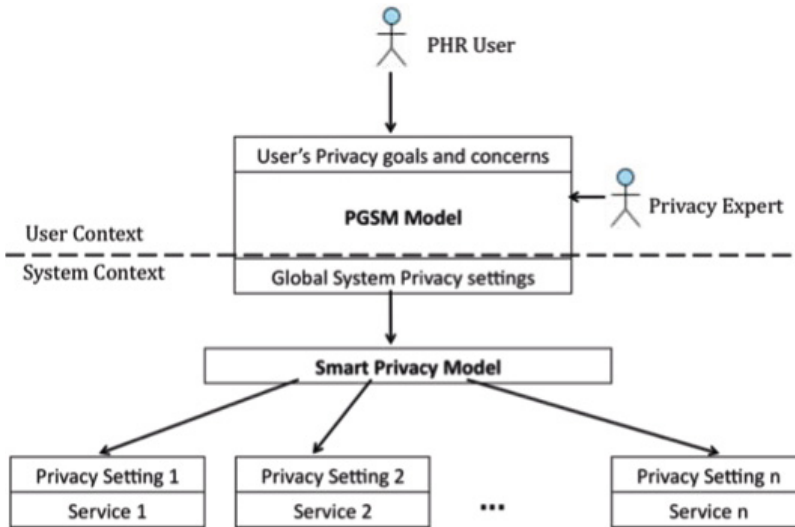


Fig. 2. Proposed privacy architecture for PHRs

In this paper we address these two problems by proposing two privacy models as shown in Fig. 2.

Privacy Goals and Settings Mediator Model (PGSM): PGSM employs i^* [8] to model the environment and the goals of participants involved in a privacy sensitive interaction. The users perceptions of their privacy are expressed in terms of goal-models of multiple actors. PGSM links the privacy features offered by a service to the high-level users' goals allowing users to understand how changes in a privacy feature affect the users goals. These goal models encode the knowledge and recommendations of privacy experts as well.

Smart Privacy Model: The output of PGSM is a set of system-level privacy settings. These settings cannot be directly utilized by the personal workflow at run-time for the privacy enforcement purpose due to the lack of semantic interoperability. Multiple services are in a choreography in a personal workflow and each one may have its own privacy enforcement mechanism. Furthermore, in a personal workflow users may


substitute a services every now and then and the new service must be able to enforce what the user had configured for the substituted service. Hence, the Smart Privacy Model is responsible to make the user privacy settings reusable and interoperable across multiple application services. The model is designed using Process Specification Language (PSL[16]). PSL allows capturing users' privacy preferences in terms of constraints over the occurrence of activities at run-time execution of personal workflows.

We now describe how two components of the proposed privacy architecture addresses the personal web privacy requirements using a motivating scenario.

3 Motivating Privacy Scenario

In a hypothetical scenario (adapted from [21], Sharing data with fitness coach), Mary is concerned with her blood pressure and wants to actively manage her health; hence she registers with a PHR service. She uses the functionalities available in the PHR platform to augment a new service (blood pressure collecting service) to her PHR. This service collects Mary's blood pressure at different point in time and stores them in Mary's PHR. After the collected data confirms Mary's fear she signs up with a new service in her PHR called disease management organization (DMO) to get help in managing her hypertension. In the sign-up process Mary opts to allow DMO to prepare a referral to a health club and consults with her fitness coach to arrange a fitness plan based on Mary's conditions.

Mary's personal workflow is depicted in Fig. 3. Her goals are clear. She wants her blood pressure to be managed in a timely manner. For this reason she opts in further sharing of her information by DMO with the health club. Nevertheless, Mary is concerned with her privacy too. When registering with the PHR platform or augmenting any of the three services mentioned above Mary is exposed to different privacy agreements and/or set of features that she has to agree or set in order to create her personal workflow.

As indicated in Fig. 3 by this icon () , there are five interactions between these services and Mary's PHR data. What Mary agrees to defines how her PHR data will be used. While Mary is concerned about her privacy, she finds it very difficult to understand these agreements. She simply accepts them in order to achieve her workflow's goals. In other words, while Mary is concerned about her privacy and does not want her data being misused, she is also concerned if her privacy settings delay her from receiving timely treatment. From the users' perspective, these are clear expectations and concerns, although not as concrete as the features and constraints that offered by the services or described in agreements. Mary's expectations and goals are described in Table 1.

In contrast to the Mary's goals, from the PHR and services perspectives, the privacy is supported by a number of features. The user is responsible to pick features as she thinks are matching her needs. Table 2 describes a number of these privacy features. We limited the privacy features to only the ones offered by DMO. By the first feature Mary asks DMO to obtain her explicit consent whenever there is an interaction between DMO and the Health club. The other three features bind the access to PHR data by the Health club to some conditions or commitments. For example, access is limited only if Mary trusts DMO through her personal experience; if DMO is a covered entity under

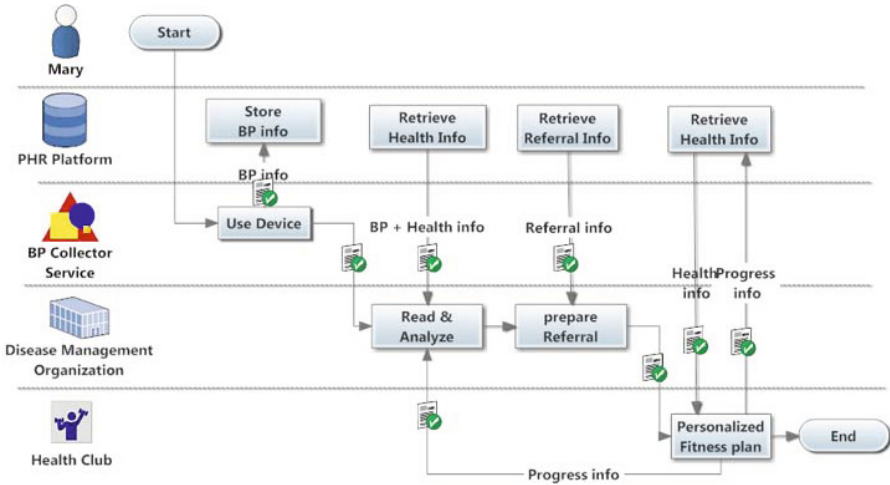


Fig. 3. Mary's personal workflow for blood pressure management

Table 1. User's goals and concerns

Title	Goals
Treatment	I want to receive treatment in an emergency case.
Timely Treatment	I am concerned if my privacy settings affect receiving treatment in a timely manner.
Privacy	I am concerned with my health data being misused.

Table 2. System privacy features

Title	Privacy Features
Explicit consent	I give explicit consent.
Personal experience	DMO is known to me.
Audit log	DMO agrees to log accesses for audit purposes.
HIPPA compliance	DMO is HIPPA compliant.

HIPAA legislation [22] (excluding access to the PHR from other jurisdictions); or if DMO agrees to log accesses for audit purpose.

By comparing the items in Table 1 and 2, we observe that Mary's concerns are high-level and casual while the system privacy features in Table 2 are strict and binding. Mary's concerns are expressed as goals, desires, and intentions. for example, *Timely treatment*, *Privacy protection*) and (*Treatment*). While the achievement of some of these goals (e.g. (*Treatment*) can be clearly judged, the achievement of the other goals cannot be judged based on a dichotomy of all-or-nothing.

The second observation is the gap between Mary's goals and the system privacy features. Mary can equivocally opt in or out the features described in Table 2. However,

it is unclear for her how selecting or not selecting these features may affect her goals. Mary is not able to find answer to questions such as; “what if I opt in all the available features in order to maximize my privacy protection?” or “does this setting allow my PHR being used even when I’m not accessible to provide consent?” To answer these types of questions, we propose to bridge the gap described above with a model that maps the high-level goals of a user to the low-level privacy features of that service providers offer.

4 PGSM Model

In this section we present the PGSM model and methodology through the scenario presented in Section 2. We describe elements and constructs of the i^* goal-oriented modeling [8] that we employ in order to address one aspect of the personal workflow specification (i.e. privacy and user preferences [3]). By using PGSM, we hide the complexity of privacy technical details from the users of the personal web applications by filling the gap with i^* conceptual models.

i^* provides a set of notations and constructs that can be used to model multiple actors’ interactions in the intentional level. i^* stands for distributed intentionality [8], referring to the premise that actors are social and they achieve their goals through the dependency relationships with the other autonomous actors. i^* has been designed to be used by software engineers for requirements analysis, particularly in the early stages of system design, to capture the intentions and expectations of stakeholders of a system. The i^* framework is also used in the design process in order to understand stakeholders’ expectations with the features of the system to-be.

In this section, we use i^* as a conceptual modelling technique to model the participants of a privacy sensitive interaction, their goals and dependencies. We first focus on the external dependencies of the participants. We then describe the internal decision-making rationale of each participant by constructing the goal models of each participant. Using the dependency model and the goal model together, we describe how goals of one participant can be externally attributed to the other participant’s’ goals. We describe all modeling steps through our example scenario.

4.1 Actors and the Network of Dependencies

In i^* , the *actor* (\odot) is an abstraction of an active entity that is capable of independent action. Actors can be humans, hardware and software, or combinations thereof. Actors are autonomous, social, and are attributed with motivations and intents [8]. As shown in Fig. 4, the PHR user, the PHR platform, the DMO, and the Health Club (as a secondary user) are some actors in our example scenario. A concrete actor is represented as an *Agent* (\ominus). Actors depend on each other to achieve *goals*, perform *tasks*, and furnish *resources* [8].

Goals are state of affairs that one or more actors of interest would like to achieve [8]. Goals (\square) are objectives for which there is a clear-cut criterion for their satisfaction. Manage my BP illness in Fig. 4 is a goal that the PHR user wants to achieve. However, the PHR user herself cannot achieve this goal. Therefore, she states it as an assertion that she wants DMO to make it true, without specifying how it is to be achieved. This has been expressed in the model as a directed goal-dependency relationship ($\text{---}\square\text{---}$, the letter D shows the direction) from the PHR user to the DMO. Using the dependency links, we can create a network of directed dependency relationships among actors (cf. Fig. 4).

If what two actors depend on each other is stated as an activity (or a set of activities) which defines a specific course of action, it is called a *task dependency* (\square). For example, DMO depends on the PHR platform to provide Authentication service.

If the subject of dependency is an entity (e.g. information or material object) the dependency is called a *resource dependency* (\square). The depender wants the dependee to furnish the entity so that it can be consumed as a resource. In Fig. 4, Partial PHR data is a resource that DMO depends on the PHR platform in order to acquire and utilize the user's health data. This dependency expresses the notion of linking the user's profile to a third party service in the existing PHR platforms.

Softgoals (\square) define the quality of the goal or task need to be achieved or performed. A softgoal is a goal without a clear-cut criterion for its achievement. Softgoals are satisfied to a "good-enough" degree, depending on subjective judgment of the actor and relevant evidence. The PHR user depends on the DMO for the softgoal Timely Treatment.

The dependency network helps in exploring the vulnerabilities of a depender since in each dependency relationship the dependee may fail to fulfill a goal or a task, or furnish a resource. For example, DMO becomes vulnerable to achieving the Timely Treatment goal if the PHR user fails to fulfill the PHR data resource dependency. To see how these dependencies impact a participant's goal, we need to extend our model in order to capture the internal reasoning structure of each actor. In the rest of this paper, we concentrate on the most important aspect of the model which is the interaction between the PHR user (Mary), DMO, and the Health club. Since, we are not investigating the interaction of Mary and PHR platform, we consider these two actors as one actor, Mary in PHR user's role.

4.2 Participants' Internal Rationale

The i^* framework offers a set of constructs, as described below, to capture the internal rationale of participants in an interaction. For every actor there is a boundary (the expanded area in Fig. 5) that defines the actor's attributed goals, tasks and resources and their internal relationships. From the PHR user's perspective, in any interaction where personal information is involved, two sets of goals can be identified.

Utility Goals. Utility goals are the reasons and values of an interaction. For example, managing blood pressure is the objective or the utility goal of the interaction between

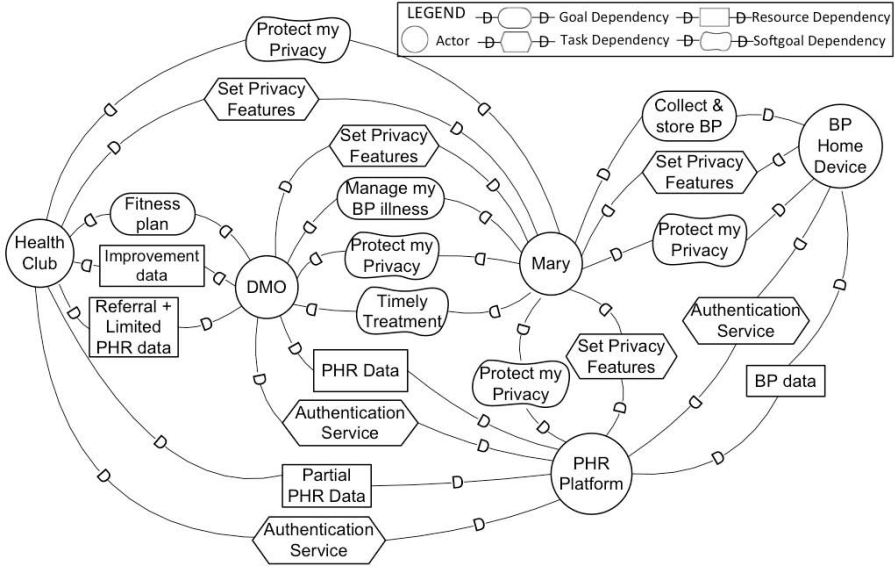


Fig. 4. Actors' dependency model

Mary and DMO. Without this goal the entire interaction would be meaningless. Thus, Mary wants to achieve this value out of the context of interaction with the DMO. In the PHR user's actor model (Fig. 5), we used the goal *Manage BP Illness* to model the utility goal.

Quality Goals. The second set consists of quality goals associated with the utility goals. In our example scenario, Mary wants her PHR data not being misused, and so, her privacy is protected. She is also concerned if her privacy setting affects the quality of her treatment. For example, if opting-in a feature causes an extra delay in managing her blood pressure illness by DMO. In the model shown in Fig. 5, we used the softgoals *Privacy* and *Timeliness* of *Treatment* to represent the PHR user's quality goals.

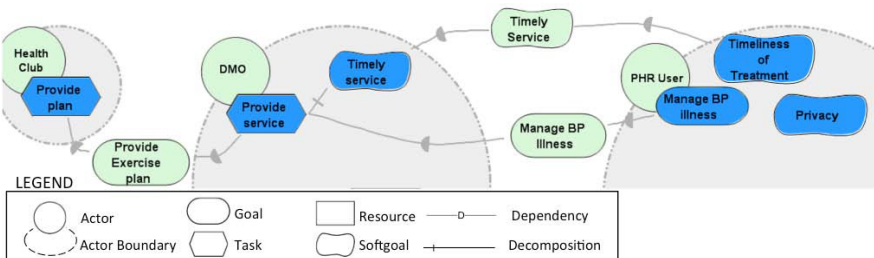


Fig. 5. Actors' internal goals

Back to the dependency network in Fig. 4 Mary depends on the DMO for both *Manage BP Illness* goal and *Timely service* softgoal. For the *Privacy* softgoal however not the external dependency, but the rationale behind the configuration of the privacy features, and the settings the user picks will determine the achievement of this goal.

Privacy Features. The privacy features, that are available to a PHR user, are modeled as tasks at the bottom of the actor’s model. For example, the privacy features *Explicit Consent* or *Audit Log* each one may impose a specific commitment or obligation to DMO that needs to be compliant with, when access to the PHR is provided or the data is utilized by DMO.

The dependency links between the PHR user and the DMO (cf. Fig. 6) represent the offered privacy options. The semantic of each privacy option determines the direction of the dependency relationship. For example, if the PHR user opts in the *Explicit Consent*, then DMO depends on the PHR user to provide consent, and consequently further actions of the DMO will be impacted by this choice subject to the internal rationale of the DMO. In contrast, if the user opts in the *Audit Log* option with two other options (*Known to me* and *Encrypt communication*) the PHR user depends on the DMO (and the PHR platform which is not modeled here) to provide the required logs and adhere to specific communication obligations.

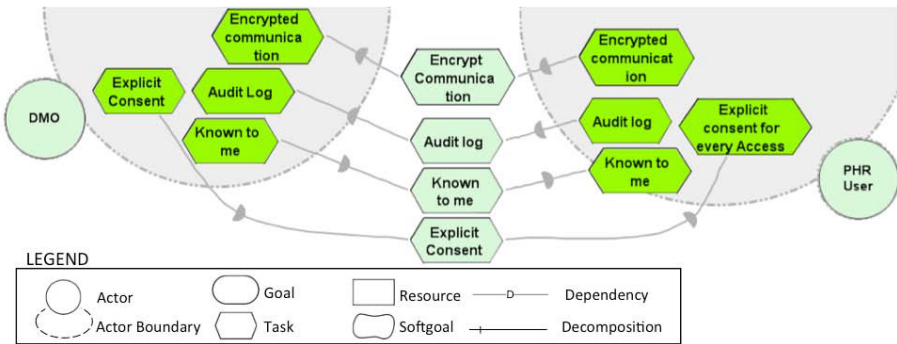


Fig. 6. Privacy features and their dependency directions

4.3 Actors’ Goal Models

To see how the selection of each feature impacts the actors’ goals, we need to extend our model to capture the internal reasoning structure of each actor. For this purpose, the *i** offers three relationship types, *Means-ends*, *Decomposition*, and *Contribution* that combined with the intentional elements introduced before (*goal*, *softgoal*, *task*, *resource*) provides the required notations for a directed goal interdependency graph. Leaf level nodes of this graph are tasks. The roots can be tasks, goals, or softgoals. The graph provides vertical traceability from the high-level concerns to the low-level tasks [22]. In PGSM, the privacy features are modeled as the tasks in the goal-model. The goal-model describes each participant’s behavior by relating the high-level goals to the

low-level privacy features. We now describe the properties of the new relationships and the goal models.

Means-ends (\rightarrow) relationship shows a particular way to achieve a goal. In our example (cf. Fig. 7), we used means-ends relationship to show that DMO as an actor in the model has different alternatives to access Mary’s partial PHR data subject to what privacy features are being opted in by the PHR user. Since not all these alternatives necessarily have the same impact on the DMO’s high-level goals, modeling these alternatives allows us to investigate the impact of each alternative.

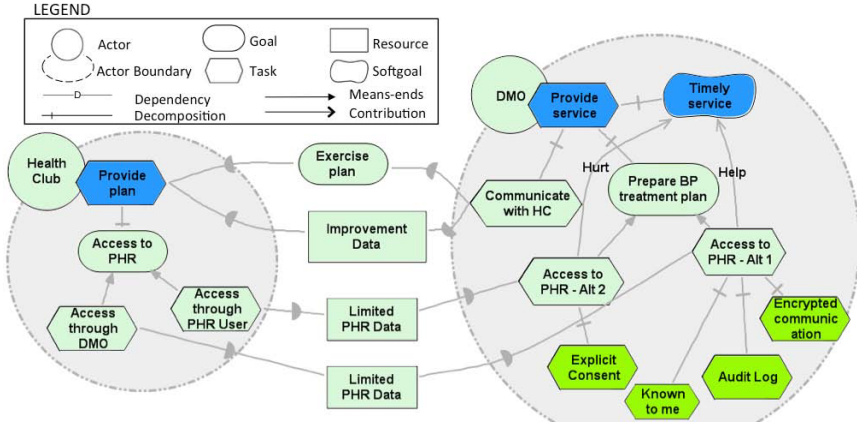


Fig. 7. Internal goal structure of each actor

Decomposition (\dashv) links are used to indicate the subtasks, subgoals, resources, and softgoals that need to be performed or satisfied in order for the parent task to succeed. In the model shown in Fig. 7, the DMO cannot achieve its goal (BP treatment) without the task communicating with HC (the Health club). Thus, we defined a top activity as Provide Service and then decompose this activity to other sub-activities such as Communicate with HC and Prepare BP treatment plan. As this example shows investigating the internal rationale of an actor may unravel new tasks, goals, and actors. We also used the decomposition link to show which privacy features need to be opted in for either of the Access to PHR tasks to be performed. For example if a user opts in Explicit Consent privacy feature, this enables the access to PHR in a specific way (i.e. Access to PHR - Alt. 2).

Contribution (\dashv) links connect tasks to softgoals or softgoals to other softgoals, indicating how the tasks contribute to achieving the actor’s quality goals. Contribution can be positive or negative, with different strengths (break, hurt, unknown, help, make). In the model shown in Fig. 7, from the DMO’s point of view not all alternative ways of accessing to the PHR data have the same impact on the Timely Service softgoal. If the access to the PHR data is bound to the Explicit Consent privacy feature, this type of access has negative contribution to the Timely Service (hurt). However, when access to the PHR data is bound to the Audit log and two other privacy features the impact is positive (Help).

Using the set of contributions, decompositions, and means-ends relationships, we were able to create the goal model for the DMO (cf. Fig. 8). With the same methodology, we construct the goal model for the PHR user based on the domain and privacy expert knowledge. In the model in Fig. 8, we combined these two goal models with the actors' dependency network- this allows the rationality of an actor being externally attributed so that the modeler can reason about impacts of other actors' behaviour [8].

4.4 Goal Model Analysis

Goal-models (cf. Fig. 8) support two types of graph-based reasoning, the forward [23] and the backward label propagation algorithms [24]. Therefore the PGSM model is capable of providing a reasoning guide for the PHR end-users to observe how changing a privacy feature may impact their privacy or utility goals.

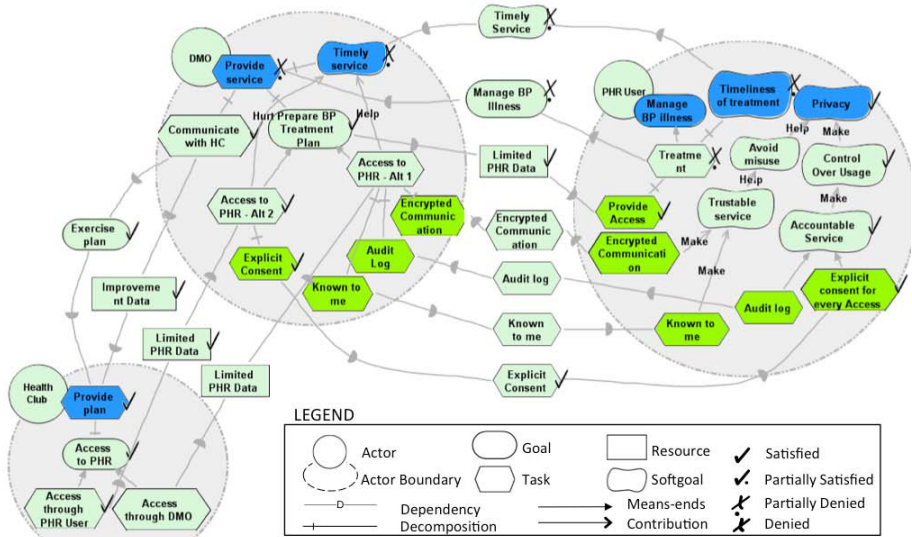


Fig. 8. Internal goal structure of each actor

Five types of qualitative *i** labels satisfied (✓), denied (✗), partially satisfied (✓/), partially denied (✗/), unknown (⊕), and conflict (⊗) are used for this purpose. When a source node receives a label, other nodes will be labeled based on the type of the relationships between the two nodes. The dependency, means-ends and decomposition relationship traverse the source label, however, the propagation of label through contribution links will be different based on the type and the strength of the link. We refer readers to [23] for more details.

In forward reasoning (bottom-up), the privacy options receive the assigned labels and then moving forward in the graph using contribution links, the degree of satisfaction or denial of the high level softgoal will be determined. The following scenario shows how the upward reasoning available in the goal graph can promote the comprehensibility of

the privacy features and in turn helps the users to make a wiser decision about their privacy settings.

Assume the PHR user fulfils the resource dependency `Provide access` and also opts in the `Explicit Consent` privacy option (note that both tasks in Fig. 8 received satisfied ✓ label). Starting from this feature with respect to the type of the out-going contribution link to the upper softgoal and to the root softgoal (i.e. `Privacy`) which is *make*, the `Privacy` softgoal is satisfied. A closer look at the dependency link from the `Explicit Consent` privacy feature in the PHR user actor model to its counterpart feature in the DMO actor model reveals a more interesting impact on the `Timeliness of Treatment` softgoal of the PHR user. Through the dependency link and then decomposition link the satisfied label (✓) propagates to a type of access to PHR (Alt. 2) that hurts the `Timely Service` softgoal (✗) in DMO actor model. This is due to the fact that for DMO, for every interaction asking for Mary’s consent and receiving PHR data through Mary makes the process stops until Mary becomes accessible. Propagating the label through the dependency relationship (`Timely Service` dependum), back to the PHR user’s goal model unravels the partial denial of the `Timely treatment` (✗) softgoal, which could be against the user’s original expectations. The denial is partial since if we check the `Partial PHR Data` dependency link between DMO and the Health club, in both alternatives, the Health club is still able to provide the exercise plan, however in one case it takes longer time for DMO to process it. Using the same analysis technique reveals that if Mary opts in the `Audit Log` option with two other features (`Known to me` and `Encrypt communication`), propagation of labels from the features through the contribution and dependency links results in partial satisfaction of `Privacy` and `Timely service` softgoals.

Having this reasoning guide accompanied by the knowledge of privacy and domain experts who create these models allow us to establish logical bridge between the privacy features that a user selects (Table 2) and the impacts on the user’s privacy and utility goals (Table 1). The judgment on selecting which privacy features could be still subjective. However, the user now have the support of privacy expert with her and more importantly can judge how change in non-understandable technical features may affect her understandable goals.

We simplified the model with having only two alternatives for privacy features; nevertheless the model allows analyzing any combination of privacy features. Furthermore, the backward reasoning can help the PHR user to find the best privacy features available given her privacy and utility goals. In the next section we demonstrate how the selected features can be utilized by multiple actors in a personal workflow for the privacy enforcement purpose at run-time.

4.5 Evaluation and Generalization of the PGSM Model

We proposed the PGSM model, using the i^* notations, in order to demonstrate and understand the alignments of the two privacy perspectives, i.e. users’ privacy perspective and systems’ privacy perspective. The example scenario, presented in this paper, is a proof of concept, aiming at demonstrating the feasibility of our work. To evaluate the methodology, we employed the method of expert interviews in the qualitative research [25]. The goals of the evaluation were to provide answers to three questions: (i) is the

comprehensibility of privacy settings a valid problem? (ii) is the PGSM model useful for privacy experts? (iii) is the settings derived by the expert using the model be of the benefit of PHR end-users?

For this purpose, we prototyped PGSM model using OpenOME [11], which is the *i** modelling and analysis tool implemented as an Eclipse Plug-in. The tool is empowered with a reasoning engine that supports both the forward and backward reasoning. The engine prompts for the human judgment whenever a non-deterministic situation arises [26]. The prototype was then presented to the three privacy analysts who had more than 10 years' experience in designing privacy policies in the health care domain. The details of evaluation process is described in [12]. The results of the evaluation showed that experts found the PGSM model useful in terms of using the model to encode their privacy knowledge. The results also showed that from the privacy experts' point of view, the model would help the PHR end-users if used during the process of privacy configuration. The ease-of-use of the model received the lowest score, suggesting that special consideration of user interface improvements needs to be in place for the model to be used by experts. The comments made by the experts when answering the open-ended questions also confirm the usability concerns.

Our context of utilizing the *i** social modelling is similar to the context of using *i** for the early requirements analysis in the process of a system design, where the *i** technique is used to align the stake holders' goals, desires and intentions with the features of a *to-be* information system. Due to this similarity, the path to the generalization of PGSM model will become easier, since the guidelines that have been designed over the years for utilizing *i** for the requirements engineering (e.g. methodologies described in *i** wiki [27] and in the other related literature such as [28] and [29]) can also be used for the design and analysis of PGSM models. Using the *i** guidelines, we describe the major steps need to be taken to generate PGSM models for the other PHR usage scenarios described in the literature (e.g. [21]). For every step, we make a reference to the PGSM model designed for the motivating scenario described in Section 3.

1. Identify participants in a scenario and model them as *i** actors (cf. Section 4.1).
2. Generate the actors' dependency model (cf. Section 4.1).
3. For each actor identify the internal rationale of being in the interaction by identifying the participant's utility and quality goals (cf. Section 4.2).
4. Use the dependency network generated in step 2 to identify from which internal goals the external dependency between actors are originated. Introduce new dependencies if it is required (cf. Section 4.2).
5. Model the privacy features as task dependency between the user and the systems (cf. Section 4.2).
6. Using the privacy experts' knowledge construct the internal goal structure of each actor (cf. Section 4.3).
7. Evaluate the PGSM model using forward or backward reasoning guide (cf. Section 4.4).

The second approach for generalization of the PGSM model is through application of privacy patterns and templates. Using patterns as an approach to facilitate *i** modelling has already been studied [30]. The PGSM models designed for the generic PHR usage scenarios can be presented as privacy templates. In the design time of a system,

these templates incorporate the privacy and domain experts' knowledge in terms of the norms applicable to a generic context (as we showed for an emergency context in our motivating scenario described in Section 3), the participants in a context and their roles, the canonical activities that occur in that context, and other concepts of the context as described by the concepts of privacy in contextual integrity [31]. When these privacy templates are used in the run-time by a PHR user, they can be personalized with two sets of user-defined parameters as described by Liaskos et. al in [32]: *values for the privacy features* and the *type and strength of contribution links* in the goal models.

When generalizing PGSM model, we are aware of the limitations we may be encountering due to the utilizing i^* as the underlying modelling notation. Although, in this thesis our goal was not to test i^* expressive power in terms of capturing the privacy requirements of an information system (as discussed in the related literature, e.g. [33], [34]), there are aspects of privacy requirements, such as ownership and custodianship of personal information, delegation of usage right, permission, and trust that could become important when bridging a user's privacy perspective with an information system's one. We discussed a number of these limitations in [35]. Further study is required to investigate if the extensions proposed in [36] and [37] for i^* to capture security and privacy requirements are applicable to PGSM when it is used to express more complex privacy scenarios.

5 Smart Privacy Model

Our goal in designing the smart privacy model is to provide seamless integration of privacy constraints in the personal workflow processes. We explain how we achieve this goal through our motivating scenario. Assume Mary, with the help of PGSM model, picks the privacy features that best satisfy her goals. Then the problem would be how she can be confident that DMO and health club will respect what she has selected. For example if Mary picks `Audit log` as a feature, the workflow processes should guarantee that all the `Health Club` communications with `DMO` are being logged in Mary's PHR. Furthermore, if Mary substitutes `DMO` in her workflow with a more valuable service or `DMO` wants to send referral to multiple health clubs, the privacy settings should have not been required to be reexamined by Mary since her goals and preferences have not been changed. The Smart Privacy Model offers a solution for this problem by offering a logical model for privacy enforcement that is interoperable and reusable among multiple services.

In this section, we first describe the smart privacy ontology, its components, and how it has been built based on the foundational privacy theory of Contextual Integrity (CI) [31]. We introduce the theory of CI as constraints on activity occurrences of a process. Thus, we describe the smart privacy ontology as an extension to the general process specification theory. We also discuss the static ontology as the second component of the smart privacy ontology. Finally, the antecedents for the privacy reasoning problems are discussed. We use the same example scenario introduced in Section 3 to describe the smart privacy ontology.

5.1 Smart Privacy Ontology

The logical framework for smart privacy is an extensible ontology that has been built using the Process Specification Language (PSL) [16] ontology. In smart privacy, theories of PSL are extended to express required privacy constructs such as pre-access conditions, post-access obligations and other communication behaviors that a workflow needs to adhere in order to respect users' privacy. As shown in Fig. 9, two main components of the Smart Privacy model are the deontic ontology and the static ontology. The deontic ontology represents all privacy constraints by extending PSL theories. Since the extension is definitional, the deontic ontology inherits properties of the consistency and entailment of PSL theories. The static ontology, described in 5.4, characterizes classes of entities and their relationships used in the Smart Privacy Ontology.

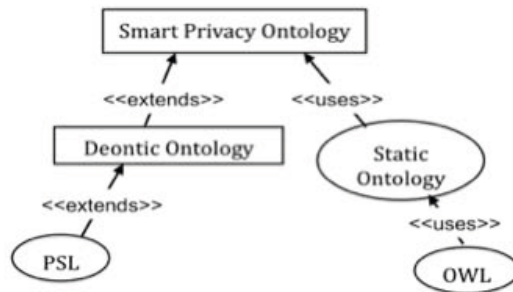


Fig. 9. Ontology-based Smart privacy

Our definition of privacy is based on the theory of Contextual Integrity (CI) [31]. CI provides a normative model and framework, for evaluating an individual's privacy when the information flows between actors [31]. The concept of *actor* in CI defines the participants in an information flow who play different roles and send and receive personal information. The concept of *principal actor* in CI represents the data subject, i.e. the participant whose personal information is at stake. *Attributes* in CI define the type of information. Two other key constructs of CI are, *contexts* and *norms*. Contexts are structured social settings characterized by the roles that actors play (e.g. Mary in a patient role), by certain values that a context offers (e.g. providing health care), the canonical *activities* and actions in which people in different *roles* perform. *Norms* prescribe and proscribe acceptable actions and practices in a particular context ([31], pp. 133-135). Based on CI theory certain patterns of flow of information in a particular context provoke the sense of privacy violation while others not [31]. The goal of the smart privacy ontology is to identify in any point in time which patterns are violating an individual's privacy and enforce the norms which are applicable to the context.

5.2 Deontic Ontology

The PSL ontology and our extended deontic ontology are a modular set of theories in the language of first order logic. In the PSL ontology, processes are described as

a certain structure of multiple activities. However, this structure might admit of many instantiations which depending on how constrained the structure is might be considerably different from one another [8]. For example in the scenario in Fig. 1, DMO performs four activities, (i) Read BP data from PHR, (ii) analyze the data, (iii) prepare the referral, (iv) and communicate the referral with the Health club. If there were not any constraints, these activities could have occurred in any order. However, this is not the case since for the workflow to deliver the functionality the only acceptable instantiation of these four activities is occurrence of the activities in a specific order as mentioned above. Therefore the PSL ontology introduces the concepts of *activity tree* and *occurrence tree* to differentiate between a structure and its instantiations.

PSL-Core [16] introduces the basic constructs to reason about activities, activity occurrences, timepoints, and objects that participate in activities. Other core theories of PSL capture the intuition for how simpler activities form a new complex activity and occurrences of its subactivities [16]. The relationship between activity and activity occurrences is represented by the *occurrence_of*(o, a) relation. The *subactivity*(a_1, a_2) relation captures the fact that a_1 is a subactivity of a_2 allowing complex activities to form. Consequently, *subactivity_occurrence*(o_1, o_2) (o_1 is a subactivity occurrence of o_2) represents the composition relation over activity occurrences. Complex activities are composed of sets of atomic activities which in turn are either primitive (i.e. they have no proper subactivities) or they are concurrent combinations of primitive activities. To capture ordering constraints over the subactivity occurrences, PSL uses the *min_precedes*(s_1, s_2, a) relation denoting that subactivity occurrence s_1 precedes the subactivity occurrence s_2 in occurrence of the complex activity a . The relation *root*(s, a) denotes that the subactivity occurrence s is the root of an activity tree for a .

Occurrences of atomic activities form the occurrence tree whose branches are equivalent to all discrete sequences of occurrences of atomic activities in the domain [16]. Although occurrence trees characterize all sequences of activity occurrences, not all of these sequences will intuitively be physically possible within a given domain. Therefore the subtree of the occurrence tree that consists only of possible sequences of activity occurrences is referred to as the legal occurrence tree. The *legal*(o) relation specifies that the atomic activity occurrence o is an element of the legal occurrence tree. The activity tree is a subtree of the legal occurrence tree characterizing the occurrences of complex activities.

In PSL, properties of the domain that change due to activity-occurrences are modeled as *fluents*. Therefore, if there is a fluent in our domain (if there is a property that changes) there must also be an activity that introduces that change. In other words, nothing changes unless there is an activity as a root cause for that change. The PSL ontology formalizes the notion of change for a domain properties in terms of occurrence of some activities. We extend this notion and show that we can use the PSL formalism to also reason about the compliance or violation of privacy in terms of changes in the properties of a *context* and its corresponding *norms* as articulated in the theory of CI [31].

5.3 Contextual Integrity as Constraints on Activity Occurrences

We formalize the concepts in CI using deontic ontology. Core to our model are activities and their occurrences. Activities are used to capture the static structure of a personal workflow. Participants (e.g. DMO and the Health club) communicate with each other by performing some activities (e.g. send PHR data). Associated with activities are the subject of privacy (i.e. Mary in our scenario) and resources (i.e. BP data in our scenario). The dynamic behaviour of a workflow is expressed by describing occurrences of activities. As activities occur and the world unfolds, elements of a context (canonical activities of a context, roles that actors play, purpose of the context) may or may not change. By precise representation of activity occurrences we are able to reason whether a context has changed or not.

The semantics of activity occurrences are also used to constrain the possible occurrences with respect to the norms of a context. The occurrence of an activity is legal (privacy is respected), if it does not violate the norms of the context that the activity belongs to. In other words, we relate the concepts of privacy compliance to the logical concept of satisfiability and entailment of legal occurrences of activities in PSL.

In addition to PSL theories, we need two sets of axioms to reason about privacy in the personal web, *context change* and *norm description*. The first set guarantees that any change in the contexts is associated with occurrences of some activities. The second set explicitly describes constraints over occurrences of such activities.

Context Change. As defined in CI, a *context* is a collective notion described by following properties: actors, roles of actors, purposes, canonical activities, and norms [31]. Expressing contexts in the Smart Privacy Model is equivalent to capturing all circumstances that may change the context properties listed above. Context's change, denoted as $\Sigma_{context}$, is a set of axioms that guarantees any change in a context's properties is associated with occurrences of some atomic activities and a context cannot change during an atomic activity occurrence. For example if over the course of the personal workflow execution, the `Health club` starts participating in an activity. According to CI, this is a change in the context, since the actor of the context has changed. Therefore, there should exist an occurrence of an activity associated with this incident. The following class of sentences formalizes the fact that when the participation of an actor in a context changes, there always exists the occurrence of an atomic activity:

$$\begin{aligned} & \textit{participates_in}(x, o_1, t_1) \wedge \neg \textit{participates_in}(x, o_1, t_2) \wedge \textit{before}(t_1, t_2) \implies \\ & \exists t_3, t_4, o_2, o_3 (\textit{sub_occ}(o_2, o_1) \wedge \textit{sub_occ}(o_3, o_1) \wedge \textit{participates_in}(x, o_2, t_3) \wedge \\ & \quad \neg \textit{participates_in}(x, o_3, t_4) \wedge \textit{next_subocc}(o_2, o_3, a)) \end{aligned}$$

In order to capture the changes in a context due to the changes in the *purpose* of a context or the *role* that an actor plays, we define two *fluents* named *for_purpose(a, p)* and *in_role(g, r, a)*. The former represents the property that the purpose for the activity *a* is *p*. The latter represents that agent *g* is playing the role *r* in the activity *a*. In the PSL ontology, fluents are changed by the occurrence of activities, and a fluent can only be changed by the occurrence of some activities. The following axiom denotes that after occurrence *o* of an activity *a* the purpose cannot change from *p*₁ to *p*₂. The axioms capturing change for the actors' roles can be written similarly.

$$\begin{aligned}
& (\forall o, a, p_1, p_2) \text{occurrence_of}(o, a) \wedge \text{prior}(\text{for_purpose}(a, p_1), o) \\
& \quad \wedge \text{hold}(\text{for_purpose}(a, p_2), o) \implies \\
& \quad (p_1 = p_2)
\end{aligned}$$

Norms. The second class of axioms in our deontic ontology represents the transmission norms that govern the privacy constraints on information flow and denoted as Σ_{norms} . There are two main classes of context norms. Norms that prohibits actions to occur if certain conditions are not satisfied, which is also sometimes called *provisions* and norms that allow actions to occur only if the agent commits to perform a set of other actions in the future, which is also called obligations [38]. PSL offers a general formula that with incorporating different temporal literals can be used to map provisions and obligations. PSL uses the process description for atomic activity to constrain the legal occurrence tree with the following general form:

$$(\forall o) \text{occurrence_of}(o, a) \wedge \text{legal}(o) \implies \varphi(o)$$

Where $\varphi(o)$ is a formula that specifies the constraints on the legal activity occurrence. In the process description this general form can be used to bind occurrence of an activity to the state that holds prior to the activity occurrence. It can be used also for other kinds of temporal preconditions that are independent to the state or when the norm implies necessity of occurrence of another activity. For example the `known to me` privacy feature in our motivating example expresses a precondition for access and can be represented as follows:

$$\begin{aligned}
& (\forall o_1) \text{occurrence_of}(o_1, \text{PHR_data_access}) \wedge \text{legal}(o_1) \implies \\
& (\exists o_2) \text{occurrence_of}(o_2, \text{previous_encounter}) \wedge (\text{earlier}(o_2, o_1) \wedge \text{legal}(o_2))
\end{aligned}$$

This axiom denotes that prior to occurrence of access to PHR data, previous encounter of Mary and DMO should have occurred.

The general form can also be used to capture obligations by incorporating the *begin_of* literal. For example, the privacy setting constraints `audit log` expresses an obligation for DMO and can be represented as the following deontic constraint. This axiom denotes that any occurrence of PHR data access activity requires occurrence of audit log activity sometimes in the future but prior to the occurrence of the access activity.

$$\begin{aligned}
& (\forall o_1) \text{occurrence_of}(o_1, \text{PHR_data_access}) \wedge \text{legal}(o_1) \implies \\
& \exists(o_2) \text{occurrence_of}(o_2, \text{audit_log}) \wedge (\text{begin_of}(o_2) > (\text{begin_of}(o_1)) \wedge \text{legal}(o_2))
\end{aligned}$$

As these examples demonstrate, we use the same semantics for expressing the *context change* and its applicable *norms*. When privacy settings are transcoded as the constraints

over the occurrences of particular activities, regardless of which service is responsible for the occurrence of an activity, the constraint implied by the set of axioms capturing the context change and its norms will be enforced, thus, supporting the interoperability of privacy settings across multiple services.

5.4 Static Ontology

The deontic ontology as described above works in the spirit of a static ontology. The static ontology, denoted by T_{static} in our model characterizes classes of entities used in the deontic ontology, their properties and their relationships. In PSL, resources that are required for an activity to occur can be specified as *objects*. For example, for the *send_data* activity to occur, two objects are required to exist at the same time prior to occurrence of this activity, a data sender as a participant and a data item as a resource. In the preceding subsection we used the concept of activity occurrence in PSL to capture the concepts of *context change* and *norms* in CI. We use the static ontology to map all classes of objects in CI that participate in the occurrence of activities. The top class in our static ontology is the PSL object class. We describe some of the subclasses of the object class below.

For a PSL activity to occur all characteristics of Participants of a context (actors in CI) need to be defined unambiguously. We have three classes of participants, *DateReceiver* (the one who receives the personal information), *DataSender* (the one who sends the personal information), and *DataSubject* (the one whose personal information is at stake). A *DateReceiver* or a *DataSender* can be further specialized to *UncoveredEntity* (entities that do not consider themselves as covered organizations under the specify privacy Act) and *CoverdEntity*. Data items, purposes, roles are also subclass of PSL object. Roles are described by a lattice. The superclass role is used to describe all possible participants' roles. Other subclasses can be used for specialized roles such as patients, researchers, and physicians. The researcher role can be further specialized to academic researcher, and so on.

The static ontology contributes to the smart privacy model by providing support for interoperability and more effective use of knowledge about contexts and their information transmission norms. This static ontology will be formulated in description logic

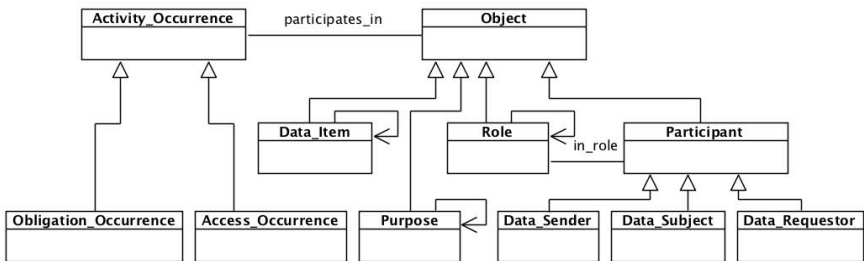


Fig. 10. Static ontology (partial)

(DL), supported by the web ontology language OWL DL [39]. Fig. 10 shows a partial representation of classes and properties in our static ontology.

5.5 Antecedents for the Reasoning Problems

We can now, relate the issues of compliance to the logical concepts of satisfiability and entailment. Entailment is key to understand whether a context complies with the information flow constraints associated with that context. Every reasoning problem in the smart privacy model has the following sets of sentences in the antecedent:

$$T_{psl} \cup T_{static} \cup \Sigma_{norms} \cup \Sigma_{context}$$

When these sets of sentences are consistent, the privacy constraints are complied with and enforced in the workflow. If this is inconsistent, then either a precondition is not satisfied or an obligation is violated. In the former case the activity has not occurred and the workflow is terminated, but in the latter the activity occurred and the query returns where and when a responsible agent failed to comply with the norms of the context.

6 Related Work

Privacy has been included in the PHR research agenda in recent years through a number of scientific surveys on privacy impacts of using PHR systems [40], [39]. The authors in [41] and [42] identified the privacy risks associated with PHR usage. They showed how privacy boundaries change when PHRs are used instead of classical medical records. Gellman et al. [41] and Wynia et al. [42] provided a set of recommendations including changes in the privacy legislations and educating PHR users in order to reduce PHRs privacy risks.

While the studies mentioned above discuss important aspects of privacy challenges in PHRs, their proposed solutions are limited to a set of recommendations. Therefore, there is a lack of tool to support users to manage their privacy in the personal web contexts in general and in PHR context in particular. In PHR systems, it is mainly a contractual agreement that guides to collection, use and disclosure of an individuals health information. In other words, in the PHR context, the freedom of individuals to have full access to their own health data, comes with the toll to take on the responsibility of self-managing their privacy when dealing with multiple services and applications. The goals of our research were providing tool support for privacy settings comprehensibility problem and proposing a framework for semantic interoperability of privacy settings.

The privacy requirements of information systems are analyzed, using conceptual models, from different perspectives. He et al. in [43] identified the importance of incorporating the privacy requirements in a system design in early stages of system development. In requirements engineering community, i* social modeling [8], after a decade of applying in a large array of information systems design, has been seen as a modeling framework to capture users privacy requirements. Using the social modeling approach, Liu et al. in [33] and Yu et al. in [34] modeled stakeholders of an information system as

actors with different privacy goals along with other quality goals such as security, accessibility and usability. All these goals are modeled as soft-goals in the *i** terminology. Then the authors analyzed, through a network of intentional dependency between actors, how achievement of these goals might be affected or how the privacy risks, threats, and vulnerabilities can be identified. Along this direction, the Secure Tropos [36], an agent-oriented security requirements engineering modeling technique, has been developed by extending *i** social modeling to address privacy requirements by including notions of ownership, permission, and delegation [37].

In the body of research introduced above, conceptual modeling is mainly used to elicit privacy requirements that need to be satisfied in the system development level. In the direction of using conceptual modelling for end-users Liaskos et al. in [32] proposed using goal-models as the overlaying reasoning structure that can be utilized by end-users in order to manage the personalization capabilities of the common personal software systems. The methodology proposed in [32] has examined on a particular email-client software to show that, using goal models, the users high level goals and preferences can be translated to configurations that satisfy those goals. While this study considered using conceptual modeling at the user level, the exploited conceptualization approach is limited to the individual goal models thus preventing the reasoning about the effects of picking one or another feature on goals of other stakeholders (social modeling) in a privacy-sensitive process to be investigated.

Compared to the privacy policy language frameworks (e.g. P3P[13], XACML [14]) and their logical counterparts (e.g. LPU[18], Privacy API[19], our work on PGSM addresses different needs. These frameworks and the preference languages built upon P3P (e.g., APPEL[44], XPref [45]) are mainly designed to express the compliance of privacy rules and regulations by an institution or a website. Besides being cumbersome to be used by an end-user, these frameworks also suffer from semantic incompatibility with the user's perspective of privacy, such as the intrinsic flexibility in a user's privacy goals. Proposals such as S4P preference language [46] addresses the flexibility of the user's privacy preferences. However, it does not offer a solution for expressing the high-level user goals. In this sense our work complements the language offered in [46]. The run-time model of our framework, the Smart Privacy ontology, compared to [14] is more expressive since it allows more complex obligations, such as users' obligation (as opposed to systems' obligations), repeating pre- and post-obligations as well as multiple responsible agents for an obligation [47] to be expressed. Our solution also uses less complex logical machinery (first order logic versus temporal logic) compared to [18]. The PSL ontology used in our ontology is highly expressive, while the PSL constructs also can be easily and systematically extended to capture more complex privacy processes.

The novelty of our research lies in alleviating the toll on the users by incorporating the knowledge of privacy experts in the decision process and reusing the privacy settings for enforcement purposes across multiple services, hence facilitating the substitution of PHR services without affecting the PHR users' privacy preferences. We also provide tool support, offering a systematic way in which a user can proactively understand the consequences of sharing as the configuration of the privacy settings change.

7 Conclusions and Future Work

This paper proposes a privacy model for the personal web applications from the user's perspective, instead of the system's perspective. We recognize that users of PHR systems are given privacy options at the system level without a clear connection to their own individual privacy goals. We identify this gap, and develop the PGSM model and methodology. PGSM provides users with a mapping between the high level user goals and low-level system privacy features.

The proposed a framework that addresses two important challenges of the personal web privacy, comprehensibility problem when privacy settings are configured by a user and interoperability of privacy settings when multiple participants are collaborating in a personal workflow. The proposed PGSM model supports the users' privacy configuration tasks, captures the experts' privacy knowledge in a particular PHR information-sharing context. Thus, the model allows the privacy knowledge to be encoded, transferred and reused. The reasoning guide that the PGSM model offers during the usage life-cycle of an application can help PHR users to make informed privacy decisions. In this sense, the model contributes to the comprehensibility of the privacy configuration task performed by the PHR users. The initial survey results presented in [12] suggest that the PGSM model is useful to privacy experts. Furthermore, the privacy experts see value if PGSM model is used by the PHR end-users.

The second model proposed in our framework, the smart privacy model, allows the privacy settings that finally has been selected by a user to be enforced by the participants in a personal workflow. The model supports interoperability and reusability of privacy settings among multiple services. This semantic interoperability allows a user substitutes one service with another in her personal workflow without requiring to express her privacy preferences repeatedly. In the future, we plan to generate privacy patterns based on the generic PHR usage scenarios using features of the PGSM and smart privacy models.

Acknowledgments. Special thanks go to Prof. Eric Yu, Prof. Michael Grüninger, and Office of the Information and Privacy Commissionaire (IPC Ontario) for their valuable help on modeling the privacy problem and evaluation of the PGSM model. Financial support from the Natural Sciences and Engineering Research Council of Canada, and from the IBM Canada Center of Advanced Studies on Collaborative Research (CAS) Privacy Award, are greatly acknowledged.

References

1. Chignell, M., Cordy, J., Ng, J., Yesha, Y. (eds.): The Smart Internet. LNCS, vol. 6400. Springer, Heidelberg (2010)
2. Mandl, K.D., Kohane, I.S.: No Small Change for the Health Information Economy. *N. Engl. J. Med.* 360(13), 1278–1281 (2009)
3. Chechik, M., Simmonds, J., Ben-David, S., Nejati, S., Sabetzadeh, M., Salay, R.: Modeling and analysis of personal web applications: A vision. In: Proc. of CASCON, vol. 10 (2010)
4. Eytan, T.: Coming to Social Media in Care Deliver Tech Demo Day: Linking Social Networks and PHRs (2011), <http://www.tedeytan.com/2011/07/28/8708>

5. Mandl, K., Simons, W., Crawford, W., Abbett, J.: Indivo: a personally controlled health record for health information exchange and communication. *BMC Medical Informatics and Decision Making* 7(1), 25 (2007)
6. Markle Foundation: Knowledge network: Survey on public opinions on the potential and privacy considerations of individually controlled electronic personal health records. Knowledge Network, Connection for Health (2008)
7. Pollach, I.: What's wrong with online privacy policies? *Commun. ACM* 50(9), 103–108 (2007)
8. Yu, E., Giorgini, P., Maiden, N., Mylopoulos, J.: *Social Modeling for Requirements Engineering*. MIT Press (2011)
9. Greenspan, S., Borgida, A., Mylopoulos, J.: A requirements modeling language and its logic. *Information Systems* 11(1), 9–23 (1986)
10. Nuseibeh, B., Easterbrook, S.: Requirements engineering: a roadmap. In: *The Future of Software Engineering*, pp. 35–46. ACM (2000)
11. OpenOME: An open-source Organization Modeling Environment (OME) (2010), <https://se.cs.toronto.edu/trac/ome/wiki>
12. Samavi, R., Consens, M.P., Topaloglou, T.: Privacy goals and settings mediator model for PHRs. In: *SocialCom/PASSAT*, pp. 1141–1146 (2011)
13. Cranor, L., Langheinrich, M., Marchiori, M., Reagle, J.: The platform for privacy preferences (P3P)1.0 specification. W3C Recommendation (2002), <http://www.w3c.org/TR/P3P/>
14. OASIS: OASIS eXtensible Access Control Markup Language v2.0 (XACML) (February 2005)
15. Backes, M., Pfitzmann, B., Schunter, M.: A toolkit for managing enterprise privacy policies. In: *Proc. ESORICS*, pp. 162–180 (2003)
16. Gruninger, M.: Ontology of the process specification language. In: *Handbook on Ontologies*, pp. 575–592 (2004)
17. Cavoukian, A.: *Privacy By Design, Take The Challeng*. Office of Information and Privacy Commissioner of Ontario (2009)
18. Barth, A., Datta, A., Mitchell, J.C., Nissenbaum, H.: Privacy and contextual integrity: Framework and applications. In: *Proc. SP*, pp. 184–198 (2006)
19. May, M.J., Gunter, C.A., Lee, I.: Privacy APIs: Access Control Techniques to Analyze and Verify Legal Privacy Policies. In: *CSFW*, pp. 85–97. IEEE Computer Society (2006)
20. Pollach, I.: Online privacy as a corporate social responsibility: an empirical study. *Business Ethics: A European Review* 20(1), 88–102 (2011)
21. HL7 International: Consent directive use cases. online by Community-Based Collaborative Care (2008), http://wiki.hl7.org/index.php?title=Consent_Directive_Use_Cases
22. Van Lamsweerde, A.: Goal-oriented requirements engineering: A guided tour. In: *RE*, pp. 249–262. IEEE (2001)
23. Giorgini, P., Mylopoulos, J., Nicchiarelli, E., Sebastiani, R.: Reasoning with goal models. In: *Conceptual Modeling-ER 2002*, pp. 167–181 (2003)
24. Sebastiani, R., Giorgini, P., Mylopoulos, J.: Simple and minimum-cost satisfiability for goal models. In: *Persson, A., Stirna, J. (eds.) CAiSE 2004*. LNCS, vol. 3084, pp. 20–35. Springer, Heidelberg (2004)
25. Kvale, S., Brinkmann, S.: *Interviews: Learning the craft of qualitative research interviewing*. Sage Publications (2008)
26. Horkoff, J., Yu, E.: Finding solutions in goal models: an interactive backward reasoning approach. In: *Parsons, J., Saeki, M., Shoval, P., Woo, C., Wand, Y. (eds.) ER 2010*. LNCS, vol. 6412, pp. 59–75. Springer, Heidelberg (2010)

27. Grau, G., Horkoff, J., Yu, E., Abdulhadi, S.: I star guide (2010), http://istar.rwth-aachen.de/tiki-view_articles.php
28. Horkoff, J.: Iterative, Interactive Analysis of Agent-goal Models for Early Requirements Engineering. PhD thesis, University of Toronto (2012)
29. Grau, G., Franch, X., Maiden, N.A.M.: Prim: An i^{*}-based process reengineering method for information systems specification. *Information & Software Technology* 50(1-2), 76–100 (2008)
30. Strohmaier, M., Horkoff, J., Yu, E., Aranda, J., Easterbrook, S.: Can patterns improve i^{*} modeling? two exploratory studies. *Requirements Engineering: Foundation for Software Quality*, 153–167 (2008)
31. Nissenbaum, H.: *Privacy in Context: Technology, Policy, and the Integrity of Social Life*. Stanford Law Books (2009)
32. Liaskos, S., Lapouchnian, A., Wang, Y., Yu, Y., Easterbrook, S.: Configuring common personal software: a requirements-driven approach. In: RE, pp. 9–18. IEEE (2005)
33. Liu, L., Yu, E., Mylopoulos, J.: Security and privacy requirements analysis within a social setting. In: RE, pp. 151–161. IEEE (2003)
34. Yu, E., Cysneiros, L.: Designing for privacy in a multi-agent world. *Trust, Reputation, and Security: Theories and Practice* (2003) 259–269
35. Samavi, R., Topaloglou, T.: Designing privacy-aware personal health record systems. In: Song, I.-Y., et al (eds.) ER Workshops 2008. LNCS, vol. 5232, pp. 12–21. Springer, Heidelberg (2008)
36. Mouratidis, H., Giorgini, P., Manson, G.: When security meets software engineering: a case of modelling secure information systems. *Information Systems* 30(8), 609–629 (2005)
37. Giorgini, P., Massacci, F., Mylopoulos, J., Zannone, N.: Modeling security requirements through ownership, permission and delegation. In: RE, pp. 167–176. IEEE (2005)
38. Hilty, M., Basin, D., Pretschner, A.: On obligations. In: di Vimercati, S.d.C., Syverson, P., Gollmann, D. (eds.) ESORICS 2005. LNCS, vol. 3679, pp. 98–117. Springer, Heidelberg (2005)
39. Halamka, J., Mandl, K., Tang, P.: Early experiences with personal health records. *Journal of the American Medical Informatics Association* 15(1), 1–7 (2008)
40. Kaelber, D., Jha, A., Johnston, D., Middleton, B., Bates, D.: A research agenda for personal health records (phrs). *Journal of the American Medical Informatics Association* 15(6), 729–736 (2008)
41. Gellman, R.: Personal health records: Why many phrs threaten privacy. Technical report, World Privacy Forum (2008)
42. Wynia, M., Dunn, K.: Dreams and nightmares: practical and ethical issues for patients and physicians using personal health records. *The Journal of Law, Medicine & Ethics* 38(1), 64–73 (2010)
43. He, Q., Antón, A., et al.: A framework for modeling privacy requirements in role engineering. In: Proc. of REFSQ, vol. 3, pp. 137–146 (2003)
44. Cranor, L., Langheinrich, M., Marchiori, M.: A P3P preference exchange language 1.0 (APPEL1.0). W3C Working Draft (2002)
45. Agrawal, R., Kiernan, J., Srikant, R., Xu, Y.: An xpath-based preference language for P3P. In: WWW, pp. 629–639 (2003)
46. Becker, M., Malkis, A., Bussard, L.: S4p: A generic language for specifying privacy preferences and policies. Technical report, Technical Report MSR-TR-2010-32, Microsoft Research (2010)
47. Ni, Q., Bertino, E., Lobo, J.: An obligation model bridging access control policies and privacy policies. In: Proc. SACMAT, pp. 133–142 (2008)