Context-aware System for Dynamic Privacy Risk Inference

Karam Bou Chaaya, Mahmoud Barhamgi, Richard Chbeir, Philippe Arnould, Djamal Benslimane

Context-aware System for Dynamic Privacy Risk Inference

Application to smart IoT environments

Karam Bou Chaaya[a,*], Mahmoud Barhamgi[b], Richard Chbeir[a], Philippe Arnould[c], Djamal Benslimane[b]

[a] *Univ Pau & Pays Adour, E2S/UPPA, LIUPPA, EA3000, Anglet, France*
[b] *Université Claude Bernard Lyon 1, LIRIS lab, Lyon, France*
[c] *Univ Pau & Pays Adour, E2S/UPPA, LIUPPA, EA3000, Mont-de-Marsan, France*

Abstract

With the rapid expansion of smart cyber–physical systems and environments, users become more and more concerned about their privacy, and ask for more involvement in the protection of their data. However, users may not be necessarily aware of the direct and indirect privacy risks they take to properly protect their privacy. In this paper, we propose a context-aware semantic reasoning system, denoted as the Privacy Oracle, capable of providing users with a dynamic overview of the privacy risks taken as their context evolves. To do so, the system continuously models, according to a proposed Semantic User Environment Modeling (SUEM) ontology, the knowledge (received by the system) about the user of interest and his surrounding cyber–physical environment. In parallel, it performs continuous reasoning over modeled information, by relying on set of privacy rules, in order to dynamically infer the privacy risks taken by the user. To validate our approach, we developed a prototype based on the semantic web tools such as OWL API, SWRL API and the inference engine Pellet. We evaluated the system performance by considering multiple use cases. Our experimental results show that the Privacy Oracle can assist users by dynamically detecting their incurred privacy risks, and by tracking, in real-time, the evolution of those risks as user context changes.

Keywords: Privacy engineering, Privacy risk, Context-aware computing, Semantic reasoning, Ontology, Internet of Things

*Corresponding author
Email addresses: karam.bou-chaaya@univ-pau.fr (Karam Bou Chaaya), mahmoud.barhamgi@univ-lyon1.fr (Mahmoud Barhamgi), richard.chbeir@univ-pau.fr (Richard Chbeir), philippe.arnould@univ-pau.fr (Philippe Arnould), djamal.benslimane@univ-lyon1.fr (Djamal Benslimane)

1. Introduction

Advances in mobile and ubiquitous computing, such as the Internet of Things (IoT), have reshaped the lives of people over the last few years. Current applications of smart IoT-enabled cyber–physical systems touch almost all aspects of our daily life including healthcare (e.g., patient and elderly monitoring), entertainment and leisure (e.g., cyber–physical games, smart entertainment spaces, social events), transportation (e.g., vehicle networks, smart highways), work (smart manufacturing and work environments), etc.

While such systems promise to ease our lives, they raise major privacy concerns for their users, as the data they collect is often privacy-sensitive, such as location of individuals, patients' vital signs, etc. In fact, collected data could be misused by the providers of such systems or even sold to interested third parties and exploited for various purposes. Privacy has received extensive attention over the last decade. Existing solutions for privacy protection vary from data anonymization, (e.g., k-Anonymity [1], l-Diversity [2]), data perturbation (e.g., differential privacy [3]), privacy-aware access control [4, 5] to encryption [6]. However, those solutions have, by and large, not been designed with the objective of involving data owners (i.e., the system users whose data is collected) in the protection of their data in mind. On the other hand, several studies (e.g., [7, 8]) showed that users are becoming more and more conscious about their privacy and willing to play an active role in controlling their data. This fact was also backed by the newly released General Data Protection Regulation (GDPR) [9], which calls for more involvement of users in the protection of their data by enabling them to control what is collected, when, by whom, and for what purposes.

Some works, e.g., [10, 11], tried to deal with this requirement by providing users with the capability to specify their privacy preferences and to accept privacy policies that enforce these preferences. Even though such works empower users by giving them an active role in specifying their preferences, they still present important limitations. First, the user may not be aware of the direct and indirect privacy risks associated with sharing his data with a consumer to correctly specify his preferences in the first place. He may simply not know what can be inferred from his data, when data bits and pieces are analyzed in isolation or combined with each other or/and with side information about the user or his surroundings (e.g., information acquired from external sources such as social networks). Second, privacy preferences are often defined in a static way, i.e., they remain unchanged no matters how user's context changes. That is, as data sensitivity changes from a context to another, static preferences fluctuate between being overprotective and under-protective leading to privacy violations.

The aim of this paper is to address the first limitation, by providing users with a global overview on the privacy risks they are taking according to their relevant contexts. This overview can raise the awareness of users, and give them the ability to make informed and meaningful data sharing decisions with

data consumers. Our work is motivated by the observation that existing legal frameworks for data protection (e.g., GDPR) might not necessarily deter data consumers from (intentionally or unintentionally) abusing the data of users. The Facebook-Cambridge Analytica scandal [12] is just one episode of a long series of data abuse scandals that happened despite the existence of binding data protection laws. Therefore, the users should be involved in controlling their data, by providing them, at the data sharing decision time, with relevant information such as the privacy risks involved in sharing their data.

In this paper, we propose a context-aware semantic reasoning system, denoted as the Privacy Oracle, capable of inferring the privacy risks that are relevant in a user's context, and providing the user with a dynamic overview of those risks as context changes. Our solution covers the privacy risks that arise when a user explicitly shares his data with data consumers (e.g., by using some smart services or apps) as well as those imposed by the surrounding cyber–physical environment and on which the user has no control such as being under a CCTV surveillance in a monitored area (e.g., airport, mall, etc.). To do so, the Privacy Oracle continuously models, using a proposed Semantic User Environment Modeling (SUEM) ontology, the knowledge (received by the system) about the user of interest and his surroundings. In parallel, it performs continuous reasoning over modeled information, by relying on set of privacy rules, in order to dynamically infer the privacy risks taken by the user. To validate our approach, we developed a prototype based on semantic web tools such as OWL API, SWRL API and the inference engine Pellet. The prototype can be easily deployed on the user smart phone. We validated the risk detection and evolution process by considering several context changes for the user. We also evaluated the system performance by considering multiple use cases. Our experimental results show that the Privacy Oracle can assist users in protecting their privacy by enabling them to infer and identify the privacy risks they incur in a certain context and by ensuring a real-time monitoring of incurred risks as context changes.

The remainder of this paper is organized as follows. Section 2 defines the key terms used in the paper. Section 3 represents the motivations, and identifies our objectives and scientific challenges. Section 4 highlights an overview of the Privacy Oracle framework, and details its components starting with the proposed ontology-based model, the privacy rules, and the followed reasoning process to infer the risks. Section 5 underlines the implementation and evaluation phases. Section 6 reviews existing ontologies for user, environment, and context modeling. Finally, Section 7 concludes and highlights future perspectives.

2. Terminology

In this section, we represent the terminology used throughout the paper to convey our different concepts. Specifically, we represent the following concepts: *data owner*, *data consumer*, *personal information*, *privacy risk*, and *privacy-sensitive information*.

***Data Owner**:* any person whose data is collected, held and processed (e.g., occupants of smart homes, monitored patients, users of social networks, etc.). This entity is known also as "*Data Subject*" (GDPR) [9].

***Data Consumer*:** any stakeholder interested in collecting and/or exploiting owners' data. Consumers are classified within two categories based on their role:

- ***Service Provider*:** generally known as a service seller, a service provider is a first-party responsible for collecting data items from owners for well-specified purposes. A provider can exploit collected data, however, as stated by the California Consumer Privacy Act (CCPA) [13], they must be contractually prohibited from disclosing personal information about the owner for any purpose other than the purpose of performing the services specified in the contract. As examples of potential providers, we can cite electricity companies in smart grids, healthcare providers in e-health, mobile application service providers, etc.

- ***Third Party*:** external entity interested in buying owners' data, or the corresponding disclosed information (i.e., if owner's data were exploited by the provider), from a principally involved party. A third party has only the rights to exploit the owner's data for the purpose specified in the contract. As examples of potential third parties, we can cite government agencies, market research companies, sales companies, etc. From a legal perspective, the service provider must notify the data owner in case of selling their data to a third party [13]. As well, a third party must not sell owner's personal information to another third party unless the owner has received explicit notice [13].

A *Data Owner* can be at the same time the *Data Consumer* and vice versa.

***Personal Information (PI)*:** any information relating to an identified or identifiable data owner (GDPR [9]). According to the US legal fields on data privacy [14], and inspired from the given NIST definition of *PI* [15], a *PI* can be classified within two categories:

- ***Personal Identifiable Information*:** any information that can be used to distinguish or trace an owner's identity, such as name, home or email address, social security number, biometric records, domain-specific information (e.g., patient number in e-health), etc.

- ***Sensitive Personal Information*:** any other sensitive information that is linked or linkable to the owner, such as location, age, marital status, activity, domain-specific information (e.g., disease, salary, social friends).

***Privacy Risk (p_r)*:** defined as a risk of disclosing one or many privacy-sensitive piece of information about a user (i.e., data owner). This disclosure could lead, in some cases, to a harmful use of the disclosed information against the user.

***Privacy-Sensitive Information (p_i)*:** defined as a personal information that could be disclosed about a user (i.e., data owner), in a specific context, when combining or/and exploiting data related directly or indirectly to the user. A commonly used classification of privacy-sensitiveness of information relies on whether the information fulfills the *Personal Information* definition or not [16].

The NIST guidelines for smart grid cybersecurity [17] has identified several potential p_i instances, including (1) User-profile information (e.g., age, disease, salary), (2) habits (e.g., daily activities), behaviors, and preferences patterns, (3) presence/absence patterns, (4) real-time surveillance, (5) appliances and medical devices used, (6) fraud detection.

3. Motivations, Objectives and Challenges

In this section, we investigate a real-life scenario to showcase some of the privacy risks that may be taken by a user when sharing her data with data consumers. Then, we describe the objective of the paper and the associated research challenges.

3.1. Running Example

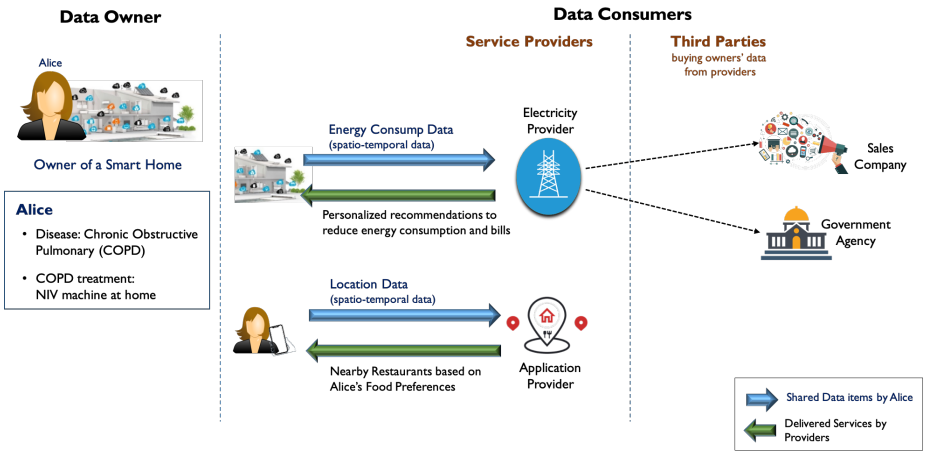

Figure 1: Motivating Scenario

Assume that Alice (cf. Fig. 1) is the owner of a smart home featuring several types of intelligent objects such as smart meters for measuring the energy consumption, smart appliances (e.g., fridge, microwave), etc. Alice is a COPD (Chronic Obstructive Pulmonary Disease) patient. She pursues her medical treatment using a deployed NIV (Non-Invasive Ventilation) machine at home. We consider at this stage the worse case scenario where Alice shares fine-grained data with the following service providers without applying any protection:

- ***Electricity provider***: Alice shares the energy consumption of her home through a deployed smart energy meter. In return, the provider provides Alice with personalized recommendations to reduce her energy consumption and bills.
- ***Application provider***: Alice shares, through a mobile application, her current location with an application service provider and gets in return the list of nearby restaurants that match with her food preferences.

Assume also that the electricity provider has signed contracts with third parties interested in exploiting the consumption data for different purposes including a publicity company and government agencies. The publicity company could be interested in analyzing the lifestyle of Alice to send her targeted advertisements (e.g., advertisement about appliances that she owns or does not own). The government agencies could be interested in identifying users involved in wrongdoing (e.g., fraud, crimes, etc.).

Alice might not be necessarily aware of the direct and indirect privacy risks she takes by sharing her data with those consumers. For example, her energy consumption (see the signature in Fig. 2) can be analyzed to infer various privacy-sensitive information about her lifestyle such as her home presence/absence hours, waking/sleeping cycles, some of her habits and activities at home (e.g., cooking, TV watching, sport activity using a treadmill) [17]. Moreover, existing works (e.g., [18]) show that the signatures of electricity consumption can be mined to identify the specific appliances (e.g., medical devices) used. This could reveal the health condition of Alice, if the usage of her NIV machine is identified.

The analysis of shared location data involves important privacy risks for Alice such as becoming under real-time surveillance (i.e., Alice can be located anytime) and the disclosure of her habits, behavior and health conditions by analyzing her trajectory patterns (cf. Fig. 3). For example, if Alice is located twice per week in a pulmonary rehabilitation center for COPD patients, then she is very likely to be a COPD patient.

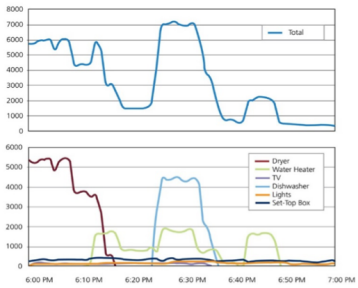

Figure 2: Analysis of an energy consumption signature

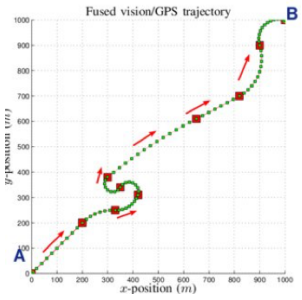

Figure 3: Analysis of a trajectory pattern

Moreover, the fact that data consumers might be able to combine multiple data items with each other or/and with other contextual information acquired from external data sources, can increase their inference capabilities, thus the sphere of possible privacy risks. For example, assume that Alice has unlawfully certified that she is living alone to be eligible for a welfare program when she submitted her application. Any data consumer (e.g., government agency) that has access to both her location and electricity consumption data can infer this fraud (it suffices to identify the usage of some specific devices such as microwaves and TVs while Alice is located outside her home).

3.2. Objectives and Challenges

The key objective of this paper is to enable the users to protect their privacy by themselves. To achieve this objective, the paper proposes an inference-based framework that would provide the users with a dynamic overview of the privacy risks they take in a given context. Such overview allows non-savvy users, e.g. Alice, to understand the implicit, direct and indirect implications of sharing their data and to make informed and meaningful data sharing decisions with data consumers (e.g., electricity company, providers of smart services such as nearby restaurants and parking spots) based on their interests and involved privacy risks. To build this dynamic context-dependent overview, we need to address the following scientific challenges:

1. ***Holistic (all-data-inclusive) privacy risk reasoning***: As discussed above, collected data can be combined with each other (e.g., electricity consumption and location metadata) and/or with other side information acquired from external data sources (e.g., profiles on social networks, public databases, etc.) to improve the inference capability of data consumers, thereby increasing the sphere of possible privacy risks. Therefore, the proposed inference framework should take into account the different data bits and pieces that are shared by the user or available to data consumers from external data sources, and explores how they combine with each other when it infers the privacy risks.

2. ***Dynamicity and context dependency of privacy risks*** : Data sensitivity and associated privacy risks may change from a context to another [19]. For example, the sensitivity of Alice's location when she is in a medical treatment center for COPD patients is higher than that when she is at home, as location in that case could be exploited to infer the health conditions of Alice. That is, as context changes, new privacy risks may emerge, while others may disappear or lose in significance. Therefore, the proposed inference framework should keep track with context changes, analyze their impacts on privacy risks and maintain an updated overview of relevant privacy risks.

3. ***Information multimodality and rich semantics***: In order to build the overview of privacy risks, the inference framework should reason on heterogeneous data pieces (i.e., data having different types and formats). These data could be also acquired from different types of data sources in the cyber–physical space such as connected IoT objects, social networks, online public databases, etc. For example, if an information ***i1*** (from a wearable connected object) indicates that Alice is located in hospital H, and another information ***i2*** (from a public database) indicates that H is dedicated to COPD treatments, then by combining ***i1*** and ***i2***, we are able to infer the presence of Alice in a COPD treatment center, which means she is likely to have a COPD condition. Therefore, the proposed inference framework should be capable of handling data pieces that are heterogeneous in terms of types, formats, origins and semantics.

4. Privacy Oracle: Context-aware System for Dynamic Privacy Risk Inference

In this section, we discuss the Privacy Oracle framework. First, we provide an overview of our proposal, followed by a formal definition of context information. Finally, we detail the Privacy Oracle modules and their components.

4.1. Approach overview

Inferring context-aware privacy risks requires first to build up a global view of the user context. This is done by gathering as much information as possible describing the user of interest, his surrounding cyber–physical environment, etc. However, collected context information can be heterogeneous (i.e., they have different data types and formats), and can be collected from different types of data sources. These data sources could be derived from both Connected environments (e.g., IoT sensor networks), and Web environments such as social networks, or any other public data source (e.g., public voting records, medical records). Moreover, gathered information may have different levels of granularity (i.e., different levels of precision). For example, the system may receive an information indicating that the user is located in a hospital, as it may receive a more precise information stating that the user is located in hospital H (i.e., specific hospital). In addition, performing in a dynamic environment that we do not control in advance makes the system unable to control or predict the knowledge to receive. Nonetheless, the system must be always capable of modeling this knowledge, and reasoning over information pieces, and the relations that exist among them, which helps in better understanding the user context, and thus the involved risks. Therefore, facing all these constraints, the decision making process of what can be inferred about the user should go beyond the classical query/retrieval schema to handle a more comprehensive semantic reasoning mechanism (e.g., deductive reasoning).

In that respect, we propose in the following a context-aware semantic reasoning system (cf. Fig. 4), denoted as the Privacy Oracle, capable of (i) analyzing the user context and dynamically inferring the involved privacy risks, and (ii) tracking, in real-time, the evolution of risks with respect to the context evolution. To do so, the Privacy Oracle continuously collects context information from the user's Connected and Web environments, models the received knowledge, and performs rule-based semantic reasoning on modeled information, while relying on a defined list of privacy rules, in order to infer the risks taken by the user.

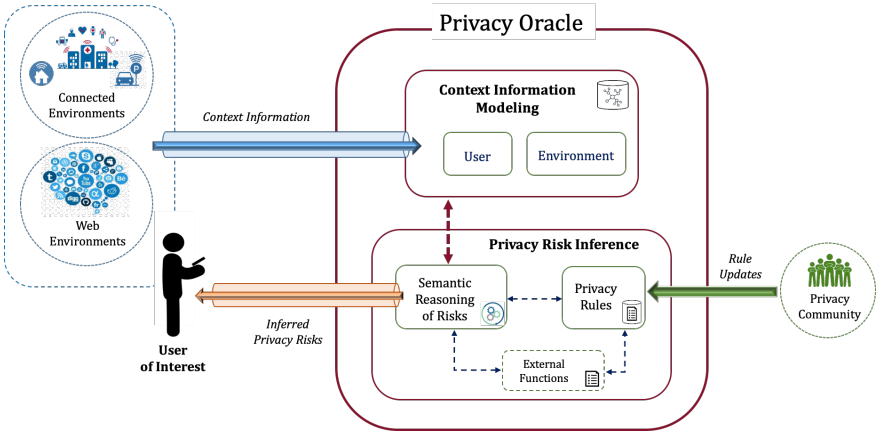

Figure 4: Privacy Oracle framework

As shown in Fig. 4, the Privacy Oracle relies on a modular framework comprised of two main modules. First, the *Context Information Modeling* module, responsible for modeling the received knowledge by the system. Then, the *Privacy Risk Inference* module, which includes two main components: (1) *Privacy Rules* component, provides a list of defined privacy rules indicating the risks to be detected by the system; (2) *Semantic Reasoning of Risks* component, provides a semantic reasoning engine responsible for inferring the user-taken risks. In regards to the *External Functions*, it serves as a function resource that can assist the two components by providing a list of external functions, available through Web-based applications (i.e., Web services), that can be used when defining the privacy rules, and during the reasoning process.

Even though we focus in this proposal on inferring the user privacy risks, the approach is re-usable and extensible. It can be adapted to various risk detection applications, and that by adapting the system components in order to handle the particularities of the application domain.

4.2. Context Information: formal definition

Several definitions were proposed in the literature for context information [20]. The mostly adopted definition, provided by Dey et al. [21], states that context information is any information that can be used to characterize the situation of the user. However, this definition remains broad, inaccurate, and non-delimited. Moreover, it does not allow to distinguish between simple information that can be extracted directly from processing a captured raw data (e.g., *user age is 25 captured from his Facebook account, home energy consumption is 2000 kW captured from deployed smart energy meter at home*), and the more complex information that can be deduced from combining and analyzing several other information (e.g., *user is located in a medical center dedicated to COPD treatments*). Therefore, we extended the given definition by proposing a two-level classification of context information, namely *elementary information*, and *complex information*. We introduced in the following a formal definition of both information levels to better stress their meaning.

Definition 1. An ***Elementary Context Information***, $\boldsymbol{e_{ci}}$, is an information generated directly from processing a captured *raw data* with its related metadata. *Raw data* instances can be captured from both Connected and Web environments. All *raw data* instances can be expressed in (1) *time* (i.e., time of capture), with each (2) being collected from a specific *data source*, and (3) describes a specific *entity*. The additional metadata elements, that may vary from a raw data type to another, are regrouped in a set of *features* associated directly to the *raw data*. Therefore, $\boldsymbol{e_{ci}}$ is defined as a 4-tuple gathering the *raw data* and its corresponding metadata aspects. It is represented as an instance of a 4-dimensional $\boldsymbol{E(CI)}$ graph regrouping all captured $\boldsymbol{e_{ci}}$ related directly or indirectly to a single user of interest (cf. Fig. 5), such that:

$$\boldsymbol{e_{ci}} : \langle\ \boldsymbol{do_i},\ \boldsymbol{t_i},\ \boldsymbol{ds_i},\ \boldsymbol{e_i}\ \rangle \text{ , where:} \tag{1}$$

- $\boldsymbol{do_i}$ is a data object regrouping the captured *raw data* with its associated set of *features*. $\boldsymbol{do_i}$ represents an instance of the $\boldsymbol{D_O}$ dimension
- $\boldsymbol{t_i}$ denotes the time of capture of *raw data*, representing an instance of the $\boldsymbol{T}$ dimension
- $\boldsymbol{ds_i}$ indicates the corresponding data source from which *raw data* was captured, representing an instance of the $\boldsymbol{D_S}$ dimension
- $\boldsymbol{e_i}$ denotes the entity described by *raw data*, representing an instance of the $\boldsymbol{E}$ dimension

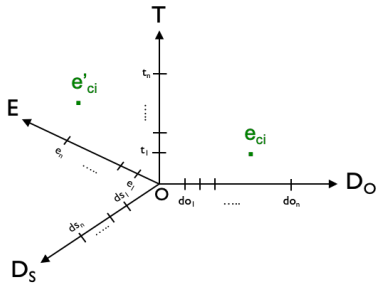

Figure 5: 4-Dimensional $E(CI)$ graph for a single user of interest

Definition 1.1. A ***Data Object*** instance, $\boldsymbol{do_i} \in \boldsymbol{D_o}$, is defined as a 2-tuple:

$$\boldsymbol{do_i}: \langle\ \boldsymbol{r_d},\ \boldsymbol{F}\ \rangle\ \text{, where:} \tag{2}$$

- $\boldsymbol{r_d}$ denotes the captured *raw data*, an unprocessed value taken directly from the data source [22]. $\boldsymbol{r_d}$ is an instance of a n-dimensional Data Property (DP) graph, where each dimension of this graph describes a specific raw data property (e.g., temperature, disease) with its corresponding *raw data* values (cf. Fig. 6 (a)).
- $\boldsymbol{F}$ denotes the set of additional features characterizing $\boldsymbol{r_d}$. Each feature value, $\boldsymbol{f_i}$, is an instance of a n-dimensional Feature Element (FE) graph, where each dimension of this graph represents a specific feature element (metadata) with its corresponding feature values (cf. Fig. 6 (b)). Therefore: $\forall \boldsymbol{f_i} \in \boldsymbol{F}, \boldsymbol{f_i} \in \boldsymbol{FE}$

Consequently, for each $\boldsymbol{r_d} \in \boldsymbol{DP}$, we have many associated $\boldsymbol{f_i} \in \boldsymbol{FE}$.

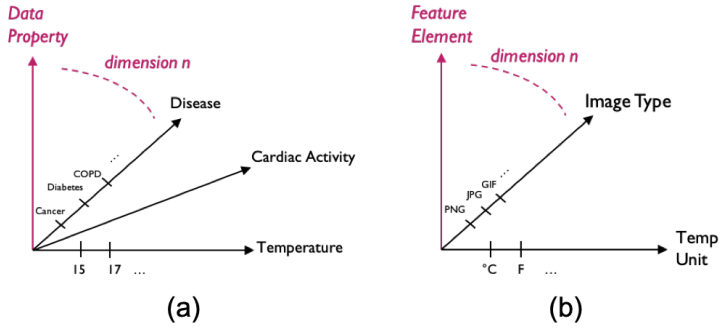

Figure 6: n-dimensional DP graph vs. n-dimensional FE graph

Definition 1.2. A ***Time*** instance, $\boldsymbol{t_i} \in \boldsymbol{T}$, is defined as a 3-tuple according to a 3-dimensional Time graph (cf. Fig. 7 (a)), such that:

$$\boldsymbol{t_i}: \langle\ \boldsymbol{date},\ \boldsymbol{time},\ \boldsymbol{ref}\ \rangle\ \text{, where:} \tag{3}$$

– ***date*** indicates the date of capture of $\boldsymbol{r_d}$

– ***time*** indicates the time of capture of $\boldsymbol{r_d}$

– ***ref*** indicates the associated date/time reference for $\boldsymbol{t_i}$

Definition 1.3. A ***Data Source*** instance, $\boldsymbol{ds_i} \in \boldsymbol{D_s}$, can derive from Connected environments (e.g., sensor network), or Web environments (e.g., social network, public data source on the Web). $\boldsymbol{ds_i}$ is defined as an instance of a n-dimensional Data Source (DS) graph (cf. Fig. 7 (b)), where each dimension represents a specific data source type (e.g., sensor network, social network).

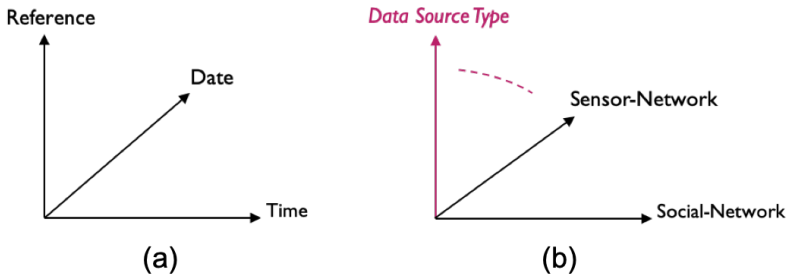

Figure 7: 3-dimensional *Time* graph vs. n-dimensional *DS* graph

Definition 1.4. An ***Entity*** instance, $\boldsymbol{e_i} \in \boldsymbol{E}$, can be the user himself, or any other entity related to the user of interest (e.g., another user, the surrounding cyber–physical environment). $\boldsymbol{e_i}$ is defined as an instance of a n-dimensional Entity graph (cf. Fig. 8), where each dimension represents a specific entity type (e.g., user, environment, system).

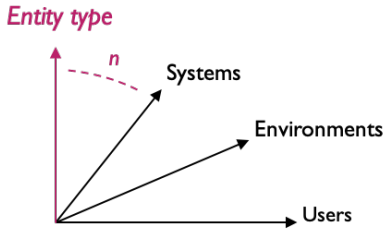

Figure 8: n-dimensional Entity graph

Definition 1.5. The ***Origine O*** of the 4-dimensional $\boldsymbol{E(CI)}$ graph is defined as 4-tuple, such that:

$$\boldsymbol{O}: \langle\, \boldsymbol{O_{Do}},\ \boldsymbol{O_T},\ \boldsymbol{O_{Ds}},\ \boldsymbol{O_E} \,\rangle \text{ , where:} \tag{4}$$

– $\boldsymbol{O_{Do}}$ represents $\boldsymbol{do_i}$ of the first captured $\boldsymbol{r_d}$
– $\boldsymbol{O_T}$ represents the date/time of capture of the first $\boldsymbol{r_d}$
– $\boldsymbol{O_{Ds}}$ represents a *null* data source instance (i.e., unknown data source)
– $\boldsymbol{O_E}$ denotes the user of interest instance

Therefore, $\boldsymbol{O}$ denotes the first raw data, captured from an unknown data source, and describing the user of interest.

Definition 1.6. The distance between any two instances of a same dimension is obtained by calling the ***Distance Function***, $\boldsymbol{distF}$, that takes as parameter the corresponding dimension, $\boldsymbol{d}$, and instances, $\boldsymbol{i_1}$ and $\boldsymbol{i_2}$, and returns the distance according to the given dimension.

$$\boldsymbol{distF}\ (\boldsymbol{d}, \boldsymbol{i_1}, \boldsymbol{i_2}) \mid \{\boldsymbol{i_1}, \boldsymbol{i_2}\} \in \boldsymbol{d} \tag{5}$$

Definition 2. A ***Complex Context Information***, $\boldsymbol{c_{ci}}$, is an information deduced from combining two or many other context information (elementary or complex). For example, by combining these two information: *'user is located in hospital H' AND 'H is dedicated to COPD treatments'*, we can deduce a new complex information $\boldsymbol{c_{ci}}$: '*user is located in a COPD treatment center*' that is privacy-sensitive for the user. Therefore, $\boldsymbol{c_{ci}}$ is defined as follows:

$$\begin{aligned} &\boldsymbol{Let} \quad \boldsymbol{E_{ci}} = \sum_{\boldsymbol{i=1}}^{\boldsymbol{n}} \boldsymbol{e_{ci}^i} \quad \boldsymbol{AND} \quad \boldsymbol{C_{ci}} = \sum_{\boldsymbol{i=1}}^{\boldsymbol{n}} \boldsymbol{c_{ci}^i} \\ &\boldsymbol{c_{ci}} \in \{\ \boldsymbol{E_{ci}} \cup \boldsymbol{C_{ci}}\ ,\ |\boldsymbol{c_{ci}}| \geqslant \boldsymbol{2}\ \} \end{aligned} \tag{6}$$

4.3. Context Information Modeling: SUEM ontology for Semantic User Environment Modeling

Facing the heterogeneity of collected information, the variety of their granularity levels, and the dynamicity of the environments, adopting a semantic data model, that maintains a flexible data structure, becomes a fundamental requirement to handle the information representation with a high-level of expressiveness. This model must be extensible and adaptable to domain-specific particularities, which makes it re-usable in many other applications.

In that respect, we propose in this paper a generic and modular ontology for Semantic User Environment Modeling, entitled SUEM[1]. This approach intro-

[1] A full documentation of the SUEM ontology can be found at: `http://spider.sigappfr.org/SUEMdoc/index-en.html`.
The ontology files are accessible on the following link: `http://spider.sigappfr.org/research-projects/suem/`.

duces concepts and properties to represent the received knowledge about users, domains of interest, and environments. As shown in Fig 9, SUEM is made of three main layers. First, the *core* layer, comprising elements to represent the generic aspects (i.e., domain-independent aspects) of both users and environments. Then, the pluggable *domain-specific user model* layer, responsible for integrating user data that are related to domain-specific applications (medical data, social data, financial data, professional data, etc.). Finally, the pluggable *domain-specific environment model* layer, that enables the alignment of the core layer with external ontologies that describe detailed components of particular environments (e.g., building, home, mall, city).

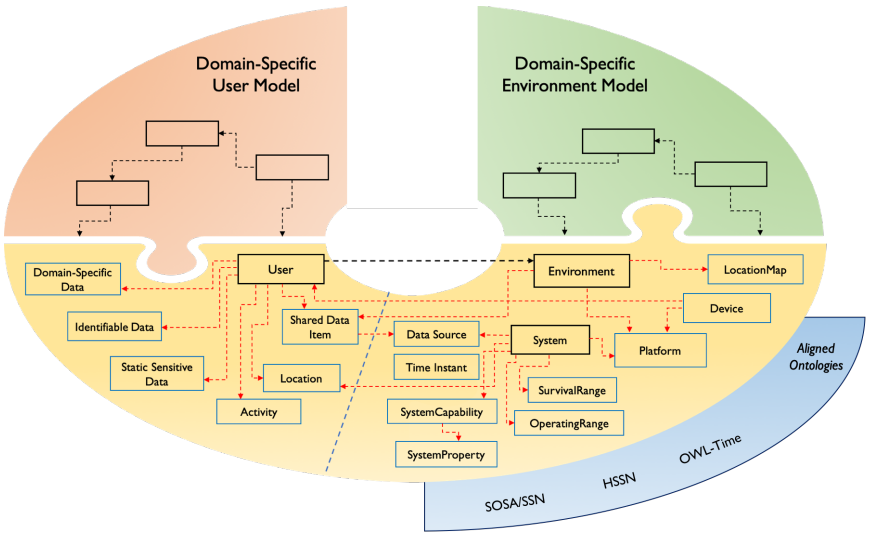

Figure 9: SUEM ontology for Semantic User Environment Modeling

Therefore, the core layer of SUEM ensures its genericity, and the pluggable layers justify its extensibility, such that it can be adaptable to any domain-specific knowledge, whether it was a user or an environment knowledge.

In the following, we discuss the core layer of the SUEM ontology. We start by detailing the proposed user ontology model. Then, we put forward the adopted environment model to represent the aspects of both static and dynamic environments. And finally, we underline the existing inter-entities relationships, and the characteristics of shared data items with consumers. We only represent the main ontology concepts and properties due to space limitation.

4.3.1. User model

A user-personal data can be either identifiable or sensitive data (cf. Section 2). As illustrated in Fig. 10, each of these two categories (i.e., identifiable

and sensitive) can regroup *Domain-Independent* (*DI*) and *Domain-Specific* (*DS*) data. *DI* data are generic data, i.e., they are not related to a particular domain (identifiable data: name, email address, etc.; sensitive data: age, preference, location, activity etc.). *DS* data are user data related to particular domains (e.g., medical data, social data, financial data, professional data). Identifiable data are almost *static* (i.e., barely change with time), however, sensitive data can be either *static* or *dynamic* (i.e., frequently change with time, such as captured sensor data describing the user location). Further examples for each data category are provided in Fig. 10.

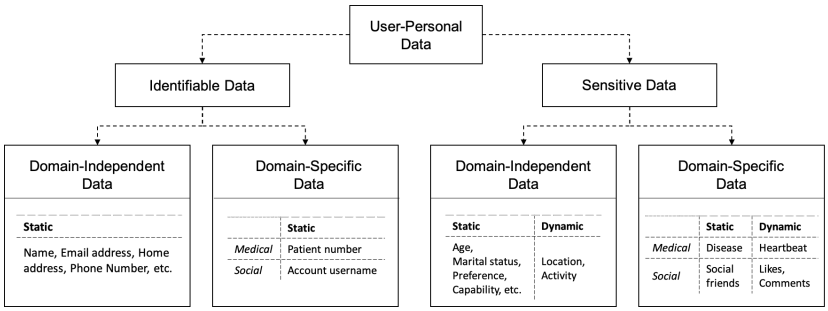

Figure 10: Proposed classification for user-personal data

The user ontology model must cover the following criteria:

1. *Generic* and *Modular* ontology, i.e., a *DI* ontology that can be re-usable in different application domains, and extensible to support the representation of the *DS* user knowledge.

2. Covers the modeling of all *DI* user data, knowing that *DS* user data are covered by the corresponding pluggable layer.

3. *Multi-modal*, considers multiple information having different data types and formats, such as scalar information, textual information, time (date-time), location (GPS coordinates or textual), etc.

4. *Multi-source*, considers information that could be acquired from different types of data sources, such as IoT sensor networks, social networks, etc.

We evaluated existing user-profile ontologies in the literature based on the aforementioned criteria (cf. Section 6). The majority of the listed approaches are domain-oriented, they were proposed for specific and well-defined purposes. None of them covers the specified needs at this stage. In that respect, we introduce in the following a generic, modular, multi-modal, and multi-source user ontology, that covers the representation of all *DI* user data, and can be easily extended to consider the corresponding *DS* data.

As shown in Fig. 11, *Identifiable Data* concept represents the user's identifiable *DI* data (e.g., name, home address, email address, etc.). Sensitive *DI* data, which could be static or dynamic, are respectively represented as follows: (1) static sensitive data are expressed through the *Static Sensitive Data* concept (e.g., age, marital status, capabilities, preferences); (2) dynamic sensitive data, i.e., location and activity data, are respectively expressed through *hssn:Location* and *Activity* concepts. A performed *Activity* by the user (e.g., running, sleeping, watching TV) has an activity time and location. The activity time can be a time instant or a time interval (e.g., user is running for 3 h). These aspects were considered by relying on the most commonly used and recommended time ontology, namely OWL-Time ontology [23], which defines a *time:TemporalEntity* concept that can be either a *time:Instant*, or a *time:Interval*.

Regarding *DS* user data, both identifiable and sensitive, the proposed model includes a *Domain-Specific Data* concept, that enables the alignment of our ontology with any other domain-centric user ontology, which justifies its extensibility.

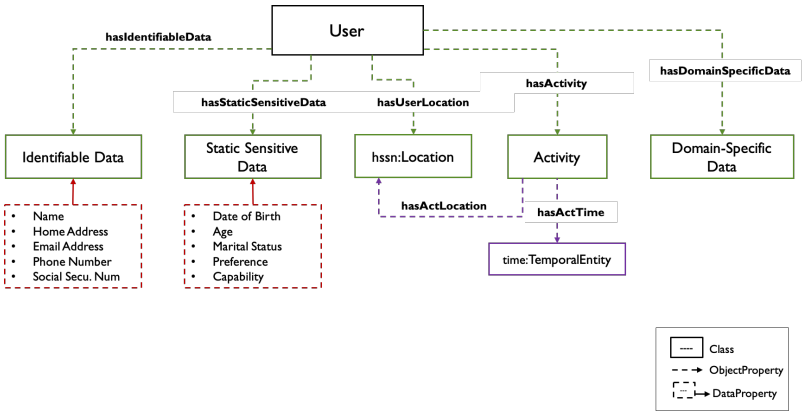

Figure 11: Proposed User Ontology

4.3.2. Environment Model

The user can be located in both (i) static environments, i.e., composed of static components (e.g., cyber–physical systems) that do not change positions with time, and (ii) dynamic environments, i.e., handle both static, and dynamic components that change positions with time such as mobile sensors and devices. This highlights the need to consider both static and dynamic aspects of an environment in the adopted model. Nonetheless, each environment (e.g., home, mall, city) can have specific aspects that do not necessarily exist for others. Therefore, we only consider in this model the common aspects between all environments, and we keep it extensible such that it can be aligned with any *DS* environment model providing concepts and properties to represent detailed

components of a particular environment.

One of the main aspects of an ontology is "Reusability". It is about reusing existing ontologies in order to avoid re-definition of same concepts. On this basis, and according to the existing environment ontologies detailed in Section 6, we decided to merge the following ontology models in order to define our environment core model: SOSA/SSN ontology [24], and HSSN ontology [25]. SOSA/SSN is a joint W3C/OGC Standard Ontology, a widely adopted and recommended ontology for describing the semantics of Sensors, Observations, Samplers and Actuation. Moreover, SOSA/SSN is a modular ontology, which respects our objective, and it respects the Ontology Design Pattern (ODP) which makes it easier to reuse/extend [26]. The HSSN ontology extends the SOSA/SSN. It extends the description of a *sosa:Plateform* to distinguish between an environment (*hssn:Infrastructure*) and a device (e.g., mobile phone, smart-watch, etc.). As well, HSSN integrates new concepts and relations to handle the dynamicity of the environments. For example, it integrates concepts to represent static and mobile sensors, the current location of systems, etc. The global environment model is illustrated in Fig. 12.

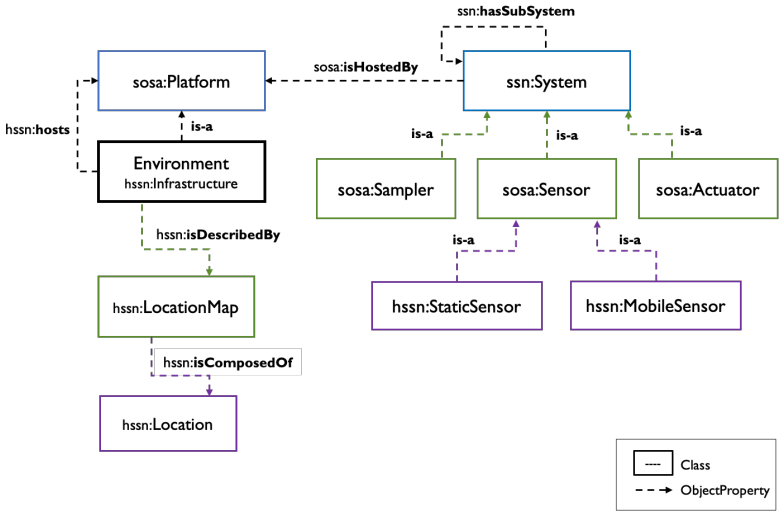

Figure 12: Environment model

4.3.3. Inter-entities relations

We show in this section how both user and environment models are interrelated in order to build the core layer of the SUEM ontology. According to Fig. 13, a *System* can be hosted by a *Platform*, which can be either an *Environment* (e.g., deployed sensors for measuring the temperature at home), or a *Device* (e.g., deployed sensors on a smart-watch). An *Environment* can host other platforms, which could be other environments (e.g., a Mall hosts several

Shops), or devices (e.g., deployed device at home). From its side, a *Device* can be attached to the *User* (i.e., user mobile devices, including wearable devices). Finally, the *User* can be located in one or many *Environment*.

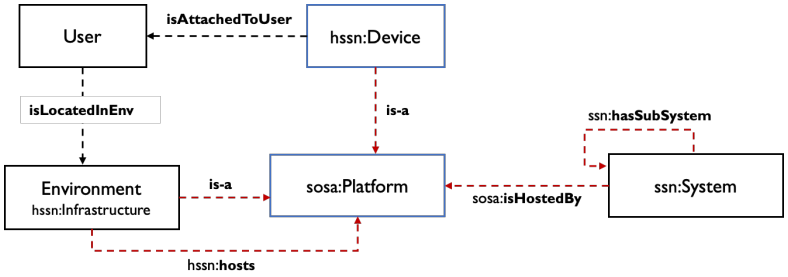

Figure 13: Inter-Entities Relationships

4.3.4. Shared Data Items

At this stage, we describe the characteristics of shared data items with data consumers. A data item can be related directly or indirectly to the user of interest. As examples of directly-related data, we can cite the user location, vital signs (medical data), motion, etc. Indirectly-related data are data describing an entity that is related directly to the user. Such an entity can be another user, an environment where the user is located, etc. As examples of indirectly-related data, we can cite the energy consumption of the user's smart home, the location of the user's wife, etc.

A same data item instance can be shared with several data consumers, where for each user/consumer connection we have a specific sharing status: (1) specific data sharing protocol(s) used to transfer data from data sources to consumers (e.g., HTTP[2], MQTT[3], CoAP[4], etc.), (2) protection mechanism(s) that may be applied on the shared data item (e.g., differential privacy mechanism, path confusion mechanism for location protection), (3) data source (e.g., sensor, social network) through which the data item is shared with this consumer. Therefore, a same data item instance can have one or many sharing status instances depending on the number of consumers with whom it is shared. All these aspects are expressed as concepts and properties in the SUEM core layer (cf. Fig. 14).

[2]Hypertext Transfer Protocol (HTTP): a web-based protocol used for transfering data through the Web.

[3]Message Queue Telemetry Transport (MQTT): a lightweight protocol for sending simple data flows from sensors to applications and middleware.

[4]Constrained Application Protocol (CoAP): a Web transfer standard protocol, intended for use in resource-constrained internet devices, such as sensor networks' nodes.

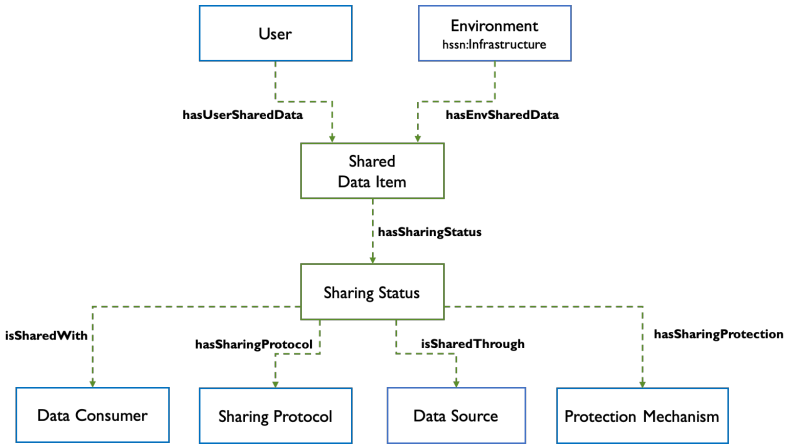

Figure 14: Shared Data Item properties

4.4. Privacy Risk Inference

In this section, we discuss the main components of the Privacy Risk Inference module, namely the *Privacy Rules* and *Semantic Reasoning of Risks* components.

4.4.1. Privacy Rules

Context Information are received and modeled by the system in real-time. However, the reasoning process requires to rely on a reference schema containing a list of privacy rules, where each rule represents a privacy risk to be detected by the system.

A ***Privacy Rule***, $\boldsymbol{PR}$, is a semantic rule indicating which combination of context information that, if captured together, could lead to infer privacy-sensitive information about the user. On this basis, we propose in the following a ***generic privacy rule syntax*** to be used in order to define a rule:

$$\boldsymbol{PR}: \ \varphi(e_{ci}) \longrightarrow \boldsymbol{P}$$

$$\text{Where}: \begin{cases} \varphi\ (a) \ = \ a_1\ \theta\ a_2\ \theta\ \ldots\ \theta\ a_n \ \mid \ n \in \mathbb{N}^* \\ \boldsymbol{P} \ = \ \langle\ p_i^1\ ;\ p_i^2\ ;\ \ldots\ ;\ p_i^k\ \rangle \ \mid \ k = \text{number of relevant } p_i \end{cases}$$

The antecedent ($\boldsymbol{\varphi(e_{ci})}$) is composed of a sequence of *elementary context information* ($\boldsymbol{e_{ci}}$) interlinked via *operators*, $\boldsymbol{\theta}$. An $\boldsymbol{e_{ci}}$ instance can be expressed as:

- Inter-related ontology concepts, expressed here using Description Logics (*DL*) [27]:

  $$\begin{aligned} &\textit{RAW-DATA} \sqsubseteq \text{isCapturedFrom.DataSource} \\ &\sqcap\ \text{hasTimeOfCapture.TimeInstant} \\ &\sqcap\ \text{isDescribingEntity.(User} \sqcup \text{Environment} \sqcup \text{System)} \\ &\sqcap\ \text{has}\textit{FEATURE.VALUE} \\ &\text{User} \sqsubseteq \neg\ \text{(Environment} \sqcup \text{System)} \\ &\text{Environment} \sqsubseteq \neg\ \text{System} \end{aligned}$$

  Where:

  - *RAW-DATA* is an individual representing a user or an environment-related data, such that:

    $$\begin{aligned} &\text{(User} \sqcup \text{IdentifiableData} \sqcup \text{StaticSensitiveData} \\ &\sqcup\ \text{Location} \sqcup \text{Activity} \sqcup \text{DomainSpecificData} \\ &\sqcup\ \text{SharedDataItem} \sqcup \text{Environment} \sqcup \text{System} \\ &\sqcup\ \text{Device} \sqcup \text{LocationMap} \sqcup \text{SystemCapability} \\ &\sqcup\ \text{SystemProperty} \sqcup \text{OperatingRange} \\ &\sqcup\ \text{SurvivalRange)}\ (\textit{RAW-DATA}) \end{aligned}$$

  - has*FEATURE.VALUE* denotes a specific *feature element* characterizing *RAW-DATA* (e.g., hasTemperatureUnit, hasImageType), with its corresponding *feature value* (cf. Eq. 2)

- An external function call, that takes as parameter many existing information, and returns new derived context information that could be related directly or indirectly to the user of interest. The list of available external functions is provided through the *External Functions* component (cf. Fig. 4).

An ***operator***, $\boldsymbol{\theta}$, can be any operator allowing the inter-linking of two or many $\boldsymbol{e_{ci}}$ instances. In this study, we focused on using the following operators:

- Semantic operators (e.g., ontology properties)

– Logical operators (e.g., AND, OR)

– Temporal operators (e.g., before, after, overlaps, during)

– Spatial operators (e.g., inside, outside, intersect, equal)

The consequent ($\boldsymbol{P}$) is composed of a sequence of atoms indicating the corresponding set of privacy-sensitive information ($\boldsymbol{p_i}$) to be disclosed about the user (cf. Section 2). Each $\boldsymbol{p_i}$ instance is represented through ontology concepts with respect to the following *DL* syntax:

$$\text{PrivacySensitiveInfo} \sqsubseteq \exists\ \text{hasDescription}.\mathit{PSI}$$

Where:

- *PSI* is a String value denoting the description of $\boldsymbol{p_i}$ (e.g., presence/absence patterns, real-time surveillance, fraud detection, disease inference).

We provide in the following two examples of potential privacy rules defined according to the proposed syntax:

– Rule-1: A user is sharing his location data with a data consumer. This raises the risk of being subject to real-time surveillance:

$\boldsymbol{PR_1}$ **:** User(?u) *AND* SharedDataItem(LOCATION)
AND hasUserSharedData(?u, LOCATION)
$\longrightarrow$ PrivacySensitiveInfo(PSI-1)
AND hasDescription(PSI-1, "Real-time surveillance")

– Rule-2: A user is sharing the energy consumption data of his smart home. This raises the risk of disclosing the presence and absence hours of the user at home by any consumer having access to this data:

$\boldsymbol{PR_2}$ **:** User(?u) *AND* Environment(?e) *AND livesIn*(?u, ?e)
AND SharedDataItem(ENERGY-CONSUMP)
AND hasEnvSharedData(?e, ENERGY-CONSUMP)
$\longrightarrow$ PrivacySensitiveInfo(PSI-2)
AND hasDescription(PSI-2, "Presence and absence patterns")

A defined privacy rule can be classified within two categories: (1) *domain-independent rule*, i.e., imported independently from the application domain; (2) *domain-specific rule*, i.e., imported only if it meets the addressed domain of

application, such as rules related to the healthcare domain, smart grid domain, etc. For example, $\boldsymbol{PR_1}$ is a domain-independent rule, however, $\boldsymbol{PR_2}$ is related to the smart home and smart grid domains, so it only exists if one of these domains is considered in the application. Additional examples of potential privacy rules from both categories are given in Table 1.

<table>
<tr><td colspan="2">Notations:
User(?u) ; Environment (?e1) ; Environment (?e2) ; Environment (BEACH) ; ProfessionalData(SICK-LEAVE)
DiseaseData (?dis) ; SharedDataItem(LOCATION) ; SharedDataItem(ENERGY-CONSUMP)
SharedDataItem(CONTACT-SWITCHES) ; System(SURV-CAM) ; System(LIC-PLATE-RECOG)
Activity(HIGH-BATHROOM-ACT) ; TreatmentDevice(?dev)</td></tr>
<tr><td colspan="2">Domain-independent rules</td></tr>
<tr><td colspan="2">Rule-3: A user is sharing his location data and he is located in a hospital dedicated to the treatment of a specific disease.
$\boldsymbol{PR_3}$: hasUserSharedData(?u, LOCATION) AND isLocatedInEnv(?u, ?e1) AND isSpecializedInDisease(?e1, ?dis)
$\longrightarrow$ PrivacySensitiveInfo(PSI-3) AND hasDescription(PSI-3, "Disease Inference")</td></tr>
<tr><td colspan="2">Rule-4: A user is sharing his location data.
$\boldsymbol{PR_4}$: hasUserSharedData(?u, LOCATION)
$\longrightarrow$ PrivacySensitiveInfo(PSI-4) AND hasDescription(PSI-4, "Habits, behaviors, and preferences inference")</td></tr>
<tr><td colspan="2">Rule-5: A user is sharing his location data. He is located at the beach when he is on a sick leave.
$\boldsymbol{PR_5}$: hasUserSharedData(?u, LOCATION) AND (isLocatedInEnv(?u, BEACH)
DURING hasProfessionalData(?u, SICK-LEAVE))
$\longrightarrow$ PrivacySensitiveInfo(PSI-5) AND hasDescription(PSI-5, "Fraud detection")</td></tr>
<tr><td colspan="2">Rule-6: A user is located in an environment hosting surveillance cameras.
$\boldsymbol{PR_6}$: isLocatedInEnv(?u, ?e1) AND hosts(?e1, SURV-CAM) OR (hosts(?e1, ?e2) AND hosts(?e2, SURV-CAM))
$\longrightarrow$ PrivacySensitiveInfo(PSI-6) AND hasDescription(PSI-6, "Presence inference in current environment")</td></tr>
<tr><td colspan="2">Rule-7: A user is located in a smart parking hosting a license plate recognition system.
$\boldsymbol{PR_7}$: isLocatedInEnv(?u, ?e1) AND hosts(?e1, LIC-PLATE-RECOG) $\longrightarrow$ PrivacySensitiveInfo(PSI-6)</td></tr>
<tr><td colspan="2">Domain-specific rules</td></tr>
<tr><td rowspan="4">Smart Home/ Smart Grid domains</td><td>Rule-8: A user is sharing the energy consumption data of his smart home.
$\boldsymbol{PR_8}$: livesIn(?u, ?e1) AND hasEnvSharedData(?e1, ENERGY-CONSUMP)
$\longrightarrow$ PrivacySensitiveInfo(PSI-7) AND hasDescription(PSI-7, "Inferring appliances and devices used")</td></tr>
<tr><td>Rule-9: A user is sharing the energy consumption data of his smart home.
$\boldsymbol{PR_9}$: livesIn(?u, ?e1) AND hasEnvSharedData(?e1, ENERGY-CONSUMP)
$\longrightarrow$ PrivacySensitiveInfo(PSI-8) AND hasDescription(PSI-8, "Waking and sleeping patterns inference")</td></tr>
<tr><td>Rule-10: A user is sharing the energy consumption data of his smart home.
$\boldsymbol{PR_{10}}$: livesIn(?u, ?e1) AND hasEnvSharedData(?e1, ENERGY-CONSUMP)
$\longrightarrow$ PrivacySensitiveInfo(PSI-9) AND hasDescription(PSI-9, "Inferring activities in current environment")</td></tr>
<tr><td>Rule-11: A user is sharing the energy consumption data of his smart home,
and has a medical treatment device deployed at home.
$\boldsymbol{PR_{11}}$: livesIn(?u, ?e1) AND hasEnvSharedData(?e1, ENERGY-CONSUMP)
AND hasTreatmentDevice(?u, ?dev)
AND isDeployedIn(?dev, ?e1) $\longrightarrow$ PrivacySensitiveInfo(PSI-3)</td></tr>
<tr><td>Smart Home domain</td><td>Rule-12: A user is sharing the contact switches data of the bathroom door of his smart home,
and has a high bathroom activity (risk of diabetes disease inference).
$\boldsymbol{PR_{12}}$: livesIn(?u, ?e1) AND hasEnvSharedData(?e1, CONTACT-SWITCHES)
AND hasActivity(?u, HIGH-BAHTROOM-ACT) $\longrightarrow$ PrivacySensitiveInfo(PSI-3)</td></tr>
</table>

Table 1: Further examples of potential privacy rules

In this paper, we focused on providing examples of privacy rules that are mainly applied in IoT scenarios. Nonetheless, the rule-setting mechanism can be used as well to define rules that fit in scenarios related to other environments, such as social media environments. Especially that both IoT and social media environments share the majority of the user-privacy concerns, including user-profiling, identification, localization, and so forth.

The more we explore different application domains, the more we discover new information combinations, that in specific contexts or not, can generate additional privacy risks. Therefore, enhancing the quality of the detection system requires to consider as much information combinations as possible from different domains. However, discovering such combinations remains a key challenge to address. As a possible solution, we could think about an outsourcing approach that enables the collaboration with experts from the privacy community, where each group of them is specialized in a specific application domain. Consequently, the privacy rules are defined and updated by the experts, and imported by the Privacy Oracle system. Nonetheless, the challenges regarding how to manage the outsourcing solution with the privacy experts, and how to manage the rules conflicts and dependencies are explored in a future work. In this paper, we only focus on how imported privacy rules could be used by the reasoning system in order to infer the involved risks.

4.4.2. Semantic Reasoning of Risks

We discuss in this section the core of the Privacy Oracle framework, the *semantic reasoning of risks* component. Our goal is to devise a Semantic Web based reasoning system, that supports rule-based continuous reasoning over context information.

Context Information are continuously received by the user's system, nonetheless, they are permanently or temporarily stored depending on the category to which they belong. First category regroups permanently-stored information, which are (1) user-profile static information that barely change with time (i.e., identifiable and sensitive static information), (2) information describing user-daily environments and their components (e.g., home, office), and (3) information about shared data items with consumers and their corresponding sharing statuses. The second category describes temporarily-stored information, which are (i) user-profile dynamic information that frequently change with time (e.g., user locations and activities), and (ii) information about public environments where the user is temporarily located (e.g., user is located in mall M). However, the information that describe public environments (e.g., malls, hospitals, cities) are permanently stored in a side database shared between all the Privacy Oracle users, and the information relating to a specific environment are imported by the user's system only when he is located in it.

During the reasoning process, the reasoning engine should always consider both existing information that are locally stored, and the new information that are continuously received depending on the evolution of the user context. For

example, an information indicating that the user is sharing his location data is locally-stored, and must be always considered when reasoning about potential privacy risks since it can generate new risks in specific contexts, when combined with other newly-received information by the system. Therefore, as shown in Fig. 15, the *semantic reasoning engine* takes as input *modeled context information* (i.e., locally-stored and newly-received information), and the imported *privacy rules*. Then, it applies a rule-based continuous reasoning over modeled information in order to infer the privacy risks. Finally, it generates as output the inferred risks, and sends them as real-time notifications to the user.

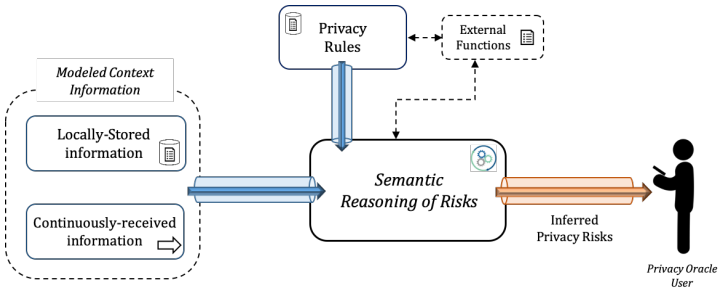

Figure 15: Semantic Reasoning of Risks

The reasoning engine reasons on modeled linked data in order to infer the user-risks. Therefore, we focus at this stage on using a semantic reasoner that supports rule-based reasoning over modeled data (e.g., Pellet OWL[5] reasoner [29]). We adapt this engine to perform a continuous reasoning, and manage the continuous data acquisition. We detail in Algorithm 1 the process followed to dynamically detect the user-taken risks.

Algorithm 1 Continuous Reasoning for Dynamic Risk Inference

1: Create Ontology instance, *onto*, mapped to *ontology.owl* file
2: Create Rule Engine instance, *ruleE*, and map it to *onto*
3: Import defined privacy rules using *ruleE*
4: Create OWL Reasoner instance, *reasoner*, and map it to *onto*
5: Execute *newInformationArrival()* function in parallel
6: **while** (true) **do**
7: Launch *reasoner* and infer the user-taken risks
8: Refresh *reasoner* to consider the ontology updates
9: Get the inferred risks' individuals from *onto*
10: Notify the user about inferred Privacy Risks
11: Save inferred updates in *ontology.owl*
12: **end while**

[5]OWL [28]: W3C semantic Web Ontology Language to represent rich and complex knowledge about things, groups of things, and relations between things.

To handle the acquisition of context information in real-time, the system supports multithreading features. It executes a second procedure, denoted as *newInformationArrival()*, in parallel with the reasoning process. This procedure is responsible for modeling the newly-arrived data according to the defined classes and properties in the ontology model. This makes the engine able to reason at the same time on both existing and newly-arrived data, and thus to analyze the user-context changes and detect the relevant risks in real-time.

5. Implementation & Evaluation

5.1. Implementation

In order to validate our approach, we implemented the Privacy Oracle throughout a Java-based prototype using OWL API, SWRL API and Pellet inference engine. The source code is available on the following link: `http://spider.sigappfr.org/research-projects/privacy-oracle/`. The objective of this experimentation is to show how the proposed system is able to dynamically infer the user privacy risks, and how it can monitor, in real-time, the evolution of those risks with respect to the evolution of the user's context.

As shown in Fig. 16, we implemented the Privacy Oracle application on the device of Alice. We considered the given context of Alice in the motivating scenario (cf. Section 3) as a basic context, and then we defined four consecutive context changes in order to monitor the evolution of the risks.

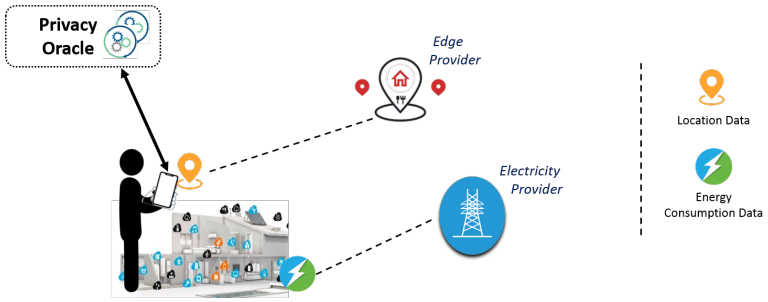

Figure 16: Privacy Oracle implementation

Context-1: *Alice* lives in *Home*, shares her *Location* data with an *Application-Provider* via a *GPSsensor* deployed on her *SmartPhone* device, shares also the *EnergyConsumption* data of her *Home* with an *ElectricityProvider* through a deployed *EnergyMeter* sensor, and has an *NVI* device deployed at *Home*. We modeled these context information according to the defined SUEM concepts as shown in Fig. 17. Regarding the medical information (i.e., disease, treatment device), we extended the user model by defining new concepts representing the user's medical data, which are related to the *domain-specific data* concept as shown in Fig. 18.

Context-2: Alice visits a shopping mall that hosts surveillance cameras.

Context-3: Alice leaves the shopping mall.

Context-4: Alice is located in a COPD treatment center.

Context-5: Alice leaves the treatment center.

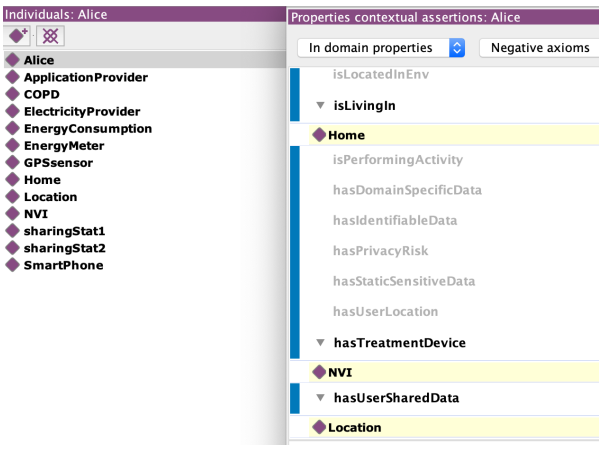

Figure 17: Context-1 related individuals

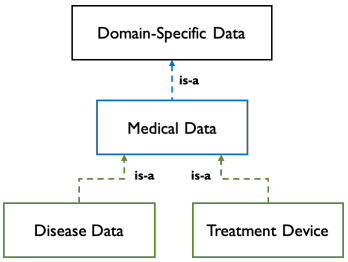

Figure 18: Medical Data concepts

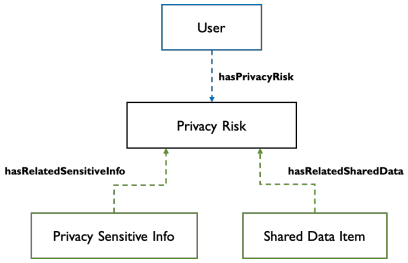

Figure 19: Privacy Risk concepts

In addition, we defined the following nine privacy rules using SWRL [30] language: $\boldsymbol{PR_1}$, $\boldsymbol{PR_2}$, $\boldsymbol{PR_3}$, $\boldsymbol{PR_4}$, $\boldsymbol{PR_6}$, $\boldsymbol{PR_8}$, $\boldsymbol{PR_9}$, $\boldsymbol{PR_{10}}$, and $\boldsymbol{PR_{11}}$. SWRL language is a W3c recommendation that provides the ability to write Horn-like semantic rules expressed in terms of OWL concepts, in order to reason about OWL individuals [30]. Therefore, SWRL represents the inferred privacy risks as OWL individuals, which requires to define their corresponding concepts and properties in the ontology model. According to Fig. 19, a *PrivacyRisk* has one or many related *PrivacySensitiveInfo*, and can have related *SharedDataItem* in case the risk was related to a shared data with a consumer. In the following, we provide an example of a defined privacy rule using the SWRL language.

$\boldsymbol{PR_1}$ using SWRL language

```
suem:User(?u)
^ suem:hasUserSharedData(?u, suem:LOCATION)
^ swrlx:createOWLThing(?prisk, 1)
^ swrlx:createOWLThing(?sens, 1)
->
suem:PrivacyRisk(?prisk)
^ suem:PrivacySensitiveInfo(?sens)
^ suem:hasDescription(?sens, "Real-time surveillance")
^ suem:hasRelatedSensitiveInfo(?prisk, ?sens)
^ suem:hasRelatedSharedData(?prisk, suem:LOCATION)
```

When mapping the defined rules to the specified context changes for Alice, we notice that seven of the nine rules are always satisfied since they are related to the basic context of Alice (i.e., Context-1). However, the eighth rule (i.e., $\boldsymbol{PR_6}$) is only satisfied in *Context-2*, and the ninth rule (i.e., $\boldsymbol{PR_3}$) is only satisfied in *Context-4*. Therefore, the system will detect seven risks for Alice in all contexts, and will monitor the evolution of the risks generated from rules $\boldsymbol{PR_3}$ and $\boldsymbol{PR_6}$.

According to the proposed reasoning algorithm (cf. Algorithm 1), the developed prototype is composed of two classes, (1) *Reasoning* class, executes a continuous reasoning process on modeled data and send as output the number of detected privacy risks with their related description (i.e., privacy-sensitive information to be disclosed) in a simple and understandable way ; (2) *newInformationArrival* class, launched in parallel with the reasoning process, it models the newly-arrived data according to the ontology model and add them in the ontology document file. On this basis, we defined in the *newInformationArrival* class the OWL individuals and properties that express the new context information to be collected by the system. These information are injected successively in the ontology file, at different time-stamps, marking the transition from a context to another. Finally, we launched the prototype, the output results are illustrated in Figs. 20–24.

Results discussion: The results show that the Privacy Oracle is able to continuously infer the involved privacy risks in the user's context, monitor their evolution, and notify the user about them in real-time. This raises the awareness of the user and enables him to take on-time precautions to protect his privacy (e.g., update his data sharing decisions depending on his context, staying/leaving decisions in case of risks raised from his surrounding environment). For example, once the system has received information indicating that Alice is in a Mall hosting surveillance cameras (Fig. 21), it has raised dynamically, and with a quasi-negligible delay of 1s, the risk of inferring her presence in the mall; this can help Alice to decide whether to stay or not in the mall, depending on if she accepts or not to take this risk.

Context 1

Date: mai 26,2019 20:30:28 ms
Number of detected privacy risks is: **7 Risks**

Risk 1:Risk of inferring Appliances and Devices used in the living environment
Risk 2:Risk of real-time remote surveillance
Risk 3:Risk of inferring Waking and Sleeping patterns
Risk 4:Risk of inferring Performed Activities in the living environment
Risk 5:Risk of inferring habits, behaviors, and preferences
Risk 6:Risk of Disease Inference from shared Energy Consumption
Risk 7:Risk of inferring presence/absence hours

Figure 20: Context 1 - Inferred Privacy Risks

Date: mai 26,2019 20:30:32 ms
Context 2: Alice is Located in a Mall hosting Surveillance Cameras

---

Date: mai 26,2019 20:30:33 ms
Number of detected privacy risks is: **8 Risks**

Risk 1:Risk of inferring Waking and Sleeping patterns
Risk 2:Risk of inferring Appliances and Devices used in the living environment
Risk 3:Risk of inferring habits, behaviors, and preferences
Risk 4:Risk of Disease Inference from shared Energy Consumption
Risk 5:Risk of inferring presence/absence hours
Risk 6:Risk of real-time remote surveillance
Risk 7:Risk of inferring Performed Activities in the living environment
Risk 8:Risk of inferring User Presence in current environment

---

Figure 21: Context 2 - Inferred Privacy Risks

Date: mai 26,2019 20:30:37 ms
Context 3: Alice leaves the mall

Date: mai 26,2019 20:30:39 ms
Number of detected privacy risks is: **7 Risks**

Risk 1:Risk of Disease Inference from shared Energy Consumption
Risk 2:Risk of real-time remote surveillance
Risk 3:Risk of inferring Performed Activities in the living environment
Risk 4:Risk of inferring Waking and Sleeping patterns
Risk 5:Risk of inferring Appliances and Devices used in the living environment
Risk 6:Risk of inferring presence/absence hours
Risk 7:Risk of inferring habits, behaviors, and preferences

Figure 22: Context 3 - Inferred Privacy Risks

Date: mai 26,2019 20:30:43 ms
Context 4: Alice is Located in a COPD treatment center

Date: mai 26,2019 20:30:44 ms
Number of detected privacy risks is: **8 Risks**

Risk 1:Risk of Disease Inference from shared Energy Consumption
Risk 2:Risk of inferring habits, behaviors, and preferences
Risk 3:Risk of Disease Inference from shared Location
Risk 4:Risk of inferring Performed Activities in the living environment
Risk 5:Risk of inferring Appliances and Devices used in the living environment
Risk 6:Risk of inferring presence/absence hours
Risk 7:Risk of inferring Waking and Sleeping patterns
Risk 8:Risk of real-time remote surveillance

Figure 23: Context 4 - Inferred Privacy Risks

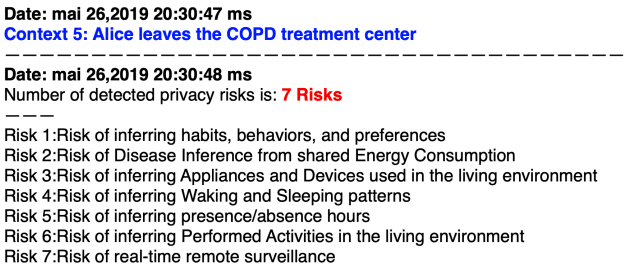

Figure 24: Context 5 - Inferred Privacy Risks

5.2. Performance Evaluation

The objective at this level is to evaluate the performance of the reasoning process. This is done by considering three use cases to study the impact factor of the following three metrics on the system's performance: (i) the number of imported privacy rules, (ii) the number of detected risks in one context, and (iii) the number of individuals over which the system reasons in order to detect the risks. The system's performance is evaluated by measuring the total execution time of one reasoning iteration. The tests were conducted on a machine equipped with an Intel i7 2.80 GHz processor and 16 GB of RAM. The chosen execution time value for each scenario is an average of 10 sequenced values.

Case 1: We focused in this use case on varying the number of imported privacy rules by the system. We fixed the number of privacy risks to detect at 100 (i.e., only 100 rules are satisfied regardless of the number of imported rules), and we ran the reasoning process five times such that: the first run scans 100 rules, the second 500, the third 1000, the fourth 5000, and the last one scans 10 000 rules . Fig. 25 shows the impact of increasing the number of imported rules on the algorithm's execution time. During each iteration, the reasoning process scans the full list of imported rules, which makes the rules number critical. Up to 1000 rules, the system is able to handle a real-time reasoning with an average execution time of 18 s per iteration, nonetheless, the execution time tends to be exponential when the rules number exceeds this threshold (e.g., 144 s in case of 5000 rules, and 405 s in case of 10 000 rules). This calls for more focus on managing the rules' import, which may vary depending on several metrics including (i) the domains of application (this justifies the proposed rules' classification in Section 4.4.1), (ii) the corresponding user of interest (i.e., a user can associate personalized weight values to the $\boldsymbol{p_i}$ elements indicating their sensitivity degree, therefore only rules leading to disclose sensitive $\boldsymbol{p_i}$ for the user are considered), and so on.

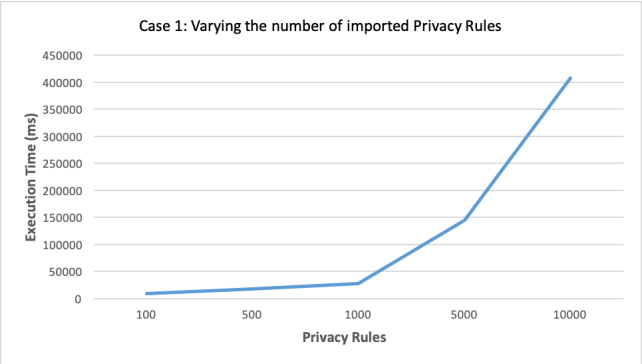

Figure 25: Case 1: varying the number of imported Rules

Case 2: We focused here on varying the number of risks to detect by the system in a single user context. We fixed the number of imported rules at 1000 (agreed threshold), and we ran the reasoning process five times such that: the first run detects 10 risks, the second 50, the third 100, then we considered 500 in the fourth run, and finally 1000 in the last one. According to Fig. 26, the total execution time remains quasi-constant for all five scenarios regardless of the number of detected risks. This is logical in view of the reasoning mechanism's operation, where imported rules are scanned one by one before generating the inferences. Therefore, the number of detected risks has no impact on the system's performance.

Figure 26: Case 2: varying the number of Risks to detect

Case 3: We focused here on varying the number of individuals over which the system reasons, in one iteration, to infer the risks. We fixed the number of rules at 1000, and the number of risks at 100, and we ran the reasoning process five times such that: in the first run, the system reasons over 50 individuals, 100 in the second run, 1000 in the third, 5000 in the fourth, and 10 000 in the last

one. As show in Fig. 27, the execution time remains quasi-constant until the number of individuals exceeds 1000, where the execution time becomes quasi-linear. Therefore, the individuals' number may have an impact on the system's performance when exceeding 1000, which justifies the need to classify stored data according to the proposed classification in Section 4.4.2 (i.e., permanently and temporarily-stored data).

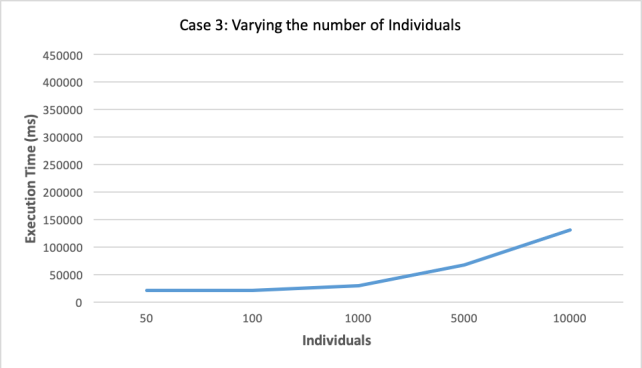

Figure 27: Case 3: varying the number of Individuals to reason on

6. Related Work: Ontology-based Models

6.1. User Modeling

A user profile is defined as a digital representation of the unique data concerning a particular user. Each of the existing ontology-based models describes the user profile in a different manner depending on the usage purpose. One of the most widely used ontologies to model people is FOAF [31]. The FOAF model is highly used in the social network field. It specifies a vocabulary that can be used to define, exchange and search for social information that describes people, their attributes and their social connections with others. Skillen et al. [32] proposed an ontological user profile modeling for context-aware application personalization within mobile environments. They described the user profile by putting forward dynamic and static aspects like health, education, capabilities, interests, preferences, and activities. Sutterer et al. [33] introduced the notion of personalized user profiles with the creation of the User Profile Ontology with Situation-Dependent Preferences Support (UPOS). The aim of this ontology was to support the situation-dependent personalization of services within changing environments by splitting the user profile into several profile subsets where each is defined in response to a specific service. Stan et al. [34] proposed an extension of UPOS ontology for situation-aware social networking, where they kept the dynamic aspects of user profiles, and considered the conjunction of context dimensions in order to better identify in real-time the situation of users. The CC/PP (Composite Capabilities/Preference Profile)

model [35], a W3C initiative, suggests an infrastructure to describe device capabilities and user preferences. CC/PP is developed specifically to facilitate the decision making process of a server, on how to customize and transfer web content to a client device in a suitable format. Skillen et al. introduced in [36] a user profile ontology based approach that provides context-aware personalized services for assisting People with Dementia (PwD) in mobile environments. They defined concepts to represent the user generic data such as personal information, location, activity, context, etc., and some of the domain-specific data such as education profile, health profile, social context, etc.

Table 2 shows a comparative study on existing user-profile ontologies with respect to our defined criteria in Section 4.3.

Criterion		Skillen et al. [32]	UPOS [33]	CC/PP [35]	Skillen et al. [36]	FOAF [31]	Stan et al. [34]
Generic (domain-independent aspects)		No	Yes	No	No	No	No
Modular / Extensible		Yes	Yes	No	Yes	No	No
Multi-modality		Yes	Yes	No	Yes	Yes	Yes
Multi-source	Connected Environments	Yes	Yes	No	Yes	No	No
	Web Environments	No	No	No	Yes	Yes	Yes
User-Generic Data	Identifiable Data	Yes	Partially [a]	No	Yes	Partially	Partially
	Sensitive Data	Partially	Partially	Partially	Partially	Partially	Partially

[a] Partially states that the related approach do not cover all user-generic identifiable or sensitive data.

Table 2: Comparative study on existing user ontology models

6.2. Environment Modeling

Domain-centric ontologies that describes specific domains impacted by IoT (domotics, agriculture, cities, etc.) are out of scope for this study since our objective is to have a generic ontology that can be re-usable in different application domains. In 2017, W3C published a new version of the most foundational ontology for sensors, the Semantic Sensor Network (SSN) Ontology [24]. The main innovation of this SSN new generation has been the introduction of the Sensor, Observation, Sample, and Actuator (SOSA) ontology, which provides a lightweight core for SSN. Thus, SOSA/SSN ontologies describe systems of sensors and actuators, samples and the process of sampling, observations, involved procedures, studied features of interest, and observed properties. Other approaches in the literature have extended the SSN ontology. However, all these works were contributed before the newly-released SSN version, so they tried to deal with the limitations of the old SSN such as the lack of description of essential IoT elements (object, actuator, service, etc.). IoT-O ontology [26] expands from old SSN with descriptions of sensors, services, units, nodes, things and actuators. It covers the following modules while applying alignments with existing ontologies: sensing (SSN), acting (SAN), life-cycle (Life-cycle), service (hRest,

MSM, wsmo-lite) and energy (PowerOnt). IoT-Lite Ontology [37] is also an instantiation of the old SSN ontology. It is a lightweight ontology that represents IoT resources, entities and services. It allows the discovery and interoperability of IoT resources in heterogeneous platforms using a common vocabulary. IoT Ontology [38] is also an expansion of the old SSN. It integrates new concepts such as *PhysicalEntity* and *SmartEntity* to support semantic expressions for interconnected, aligned and clustered entities.

6.3. Context Modeling

Various context modeling techniques exist in the literature: key–value, markup scheme, object oriented, graphical, logic based and ontology based. According to many surveys and comparative studies [20, 39], ontology-based techniques are the preferred mechanism for context modeling and reasoning in pervasive computing environments. A broad variety of ontologies and vocabularies exist to model context in smart environments. These latter are classified into two categories: (i) user-centered approaches to model human activities such as *CoBrA-ont*, *CoDAMoS*, *CONCON*, *PiVOn*, *Delivery Context Ontology*, and many others; (ii) domain-oriented approaches to describe the context and the environment where human activities occur like Location & Time ontologies (*WGS84 Geo Positioning, Time Ontology*, etc.), user profile and preferences ontologies (*FOAF, CC/PP model,* etc.), and so forth. In [40], Rodriguez et al. provided a detailed study on ontologies of both categories in terms of concepts to model and their purposes.

The main elements of the context remain neither well defined nor delimited. This led to have various dimensions of context modeling, depending on the perspectives in the field. In [41, 42], the authors have focused on the context acquired through sensors and identified four categories of context information: location, time, identity, and activity. In [43], location, time, identity and environment were considered when defining context. In [44], the authors set three main elements: user, platform and environment. *CONCON* (CONtext ONtology) [45] considered that location, user, activity and computational entity (e.g., device) are the fundamental concepts for capturing the context information. In [46], the perspective was domain-centric, the authors wanted to model healthcare context information, so they specified seven dimensions: location, individual, activity, environment, device, medical and auxiliary.

Other approaches have provided a structural representation of contexts in order to facilitate context reasoning. Schilit [47] classified the context into three categories: user contexts (user profile, location, social situation, etc.), computational contexts (network connectivity, communication costs and bandwidth, etc.), and physical contexts (lighting, noise levels, etc.). In [48], the authors underlined eight computing entity classes: user contexts (identity, preference, activity, location, etc.), device contexts (processor speed, location, etc.), application contexts (version, availability, etc.), physical environment contexts (illumination, humidity, etc.), resource contexts (availability, size, type, etc.), network

contexts (speed level, etc.), location contexts (contents, where it is subsumed, etc.), and activity contexts (start time, end time, actor, etc.).

7. Conclusion & Future Work

In this paper, we introduce a context-aware semantic reasoning approach, dubbed as the Privacy Oracle, that helps users in protecting their privacy by themselves, and that by providing them with a dynamic overview of the privacy risks they take as their context evolves. To do so, the system continuously models, according to a proposed Semantic User Environment Modeling (SUEM) ontology, the received knowledge about the user of interest and his surrounding cyber–physical environment. In parallel, it applies a holistic privacy reasoning on modeled information, by relying on set of privacy rules, in order to dynamically infer the involved privacy risks. To validate our approach, we developed a prototype based on the semantic web tools such as OWL API, SWRL API and the inference engine Pellet. We evaluated the system's performance by considering multiple use cases. Our experimental results show that the Privacy Oracle can assist users by dynamically detecting their incurred privacy risks, and by tracking, in real-time, the evolution of those risks as user context changes.

A privacy risk has an associated *probabilistic risk value* indicating its level of importance. The quantification of such value requires relying on several privacy metrics including (i) the uncertainty of the combined information pieces, (ii) the sensitivity degree of the disclosed $\boldsymbol{p_i}$ for the user, (iii) the level of trust associated by the user to the corresponding data consumer, etc. Therefore, as future work, we are investigating the quantification of privacy risks. In addition, we would like to address the challenges regarding the handling of the rules conflicts and dependencies. Finally, we aim at providing users with a dynamic list of recommended privacy protection measures to apply in order to decrease the impact of their taken privacy risks.

References

[1] L. Sweeney, "k-anonymity: A model for protecting privacy," International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, vol. 10, no. 05, pp. 557–570, 2002.

[2] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam, "\ell-diversity: Privacy beyond\kappa-anonymity," p. 24, IEEE, 2006.

[3] C. Dwork, "Differential privacy: A survey of results," in International Conference on Theory and Applications of Models of Computation, pp. 1–19, Springer, 2008.

[4] S. Oulmakhzoune, N. Cuppens-Boulahia, F. Cuppens, S. Morucci, M. Barhamgi, and D. Benslimane, "Privacy query rewriting algorithm instrumented by a privacy-aware access control model," Annales des Télécommunications, vol. 69, no. 1-2, pp. 3–19, 2014.

[5] M. Barhamgi, D. Benslimane, Y. Amghar, N. Cuppens-Boulahia, and F. Cuppens, "Privcomp: a privacy-aware data service composition system," in Proceedings of the 16th International Conference on Extending Database Technology, pp. 757–760, Citeseer, 2013.

[6] M. Barhamgi, A. K. Bandara, Y. Yu, K. Belhajjame, and B. Nuseibeh, "Protecting privacy in the cloud: Current practices, future directions," IEEE Computer, vol. 49, no. 2, pp. 68–72, 2016.

[7] J. P. Kolter, User-centric Privacy: A Usable and Provider-independent Privacy Infrastructure, vol. 41. BoD–Books on Demand, 2010.

[8] B. P. Knijnenburg, "Simplifying privacy decisions: Towards interactive and adaptive solutions.," in Decisions@ RecSys, pp. 40–41, 2013.

[9] N. Vollmer, "Table of contents EU General Data Protection Regulation (EU-GDPR)," May 2018. http://www.privacy-regulation.eu/en/index.htm.

[10] C. Castelluccia, M. Cunche, D. L. Metayer, and V. Morel, "Enhancing transparency and consent in the iot," in 2018 IEEE European Symposium on Security and Privacy Workshops (EuroS PW), pp. 116–119, April 2018.

[11] I. D. Addo, S. I. Ahamed, S. S. Yau, and A. Buduru, "A reference architecture for improving security and privacy in internet of things applications," in 2014 IEEE International Conference on Mobile Services, pp. 108–115, June 2014.

[12] "Data in the post-gdpr world," Computer Fraud & Security, vol. 2018, no. 9, pp. 17 – 18, 2018.

[13] "CCPA: California Consumer Privacy Act," 2018. https://leginfo.legislature.ca.gov/faces/billCompareClient.xhtml?bill_id=201720180AB375.

[14] G. M. Stevens, "Data security breach notification laws," tech. rep., Congressional Research Service, 2012.

[15] E. McCallister, "Guide to protecting the confidentiality of personally identifiable information (PII)," Tech. Rep. NIST SP 800-122, National Institute of Standards and Technology, Gaithersburg, MD, 2010.

[16] M. Callahan, "Us dhs handbook for safeguarding sensitive personally identifiable information," Washington, DC, 2012.

[17] V. Y. Pillitteri and T. L. Brewer, "Guidelines for smart grid cybersecurity," Tech. Rep. NISTIR 7628 Revision 1, National Institute of Standards and Technology, 2014.

[18] M. A. Lisovich, D. K. Mulligan, and S. B. Wicker, "Inferring personal information from demand-response systems," IEEE Security & Privacy, vol. 8, no. 1, 2010.

[19] M. Barhamgi, C. Perera, C. Ghedira, and D. Benslimane, "User-centric privacy engineering for the internet of things," IEEE Cloud Computing, vol. 5, no. 5, pp. 47–57, 2018.

[20] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the internet of things: A survey," IEEE communications surveys & tutorials, vol. 16, no. 1, pp. 414–454, 2014.

[21] A. K. Dey, G. D. Abowd, and D. Salber, "A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications," Human–Computer Interaction, vol. 16, no. 2-4, pp. 97–166, 2001.

[22] L. Sanchez, J. Lanza, R. Olsen, M. Bauer, and M. Girod-Genet, "A generic context management framework for personal networking environments," in 2006 Third Annual International Conference on Mobile and Ubiquitous Systems: Networking Services, pp. 1–8, July 2006.

[23] S. Cox and C. Little, "Time ontology in owl," 2017. W3C Recommendation. `https://www.w3.org/TR/owl-time/`.

[24] A. Haller, K. Janowicz, and S. Cox, "Semantic sensor network ontology," October 2017. `https://www.w3.org/TR/vocab-ssn/`.

[25] E. Mansour, R. Chbeir, and P. Arnould, "Hssn: An ontology for hybrid semantic sensor networks," In the 23rd International Database Engineering & Applications Symposium (IDEASÍ9), 2019. To appear.

[26] N. Seydoux, K. Drira, N. Hernandez, and T. Monteil, "IoT-O, a Core-Domain IoT Ontology to Represent Connected Devices Networks," in Knowledge Engineering and Knowledge Management, vol. 10024, pp. 561–576, Springer International Publishing, 2016.

[27] F. Baader, I. Horrocks, and U. Sattler, "Description logics," in Handbook on ontologies, pp. 3–28, Springer, 2004.

[28] "OWL Web Ontology Language Reference," 2004. `https://www.w3.org/TR/owl-ref/`.

[29] "Pellet - Semantic Web Standards." `https://www.w3.org/2001/sw/wiki/Pellet`.

[30] H. Ian, P.-S. Peter F., B. Harold, T. Said, G. Benjamin, and D. Mike, "Swrl: A semantic web rule language combining owl and ruleml," May 2004. `https://www.w3.org/Submission/2004/SUBM-SWRL-20040521/`.

[31] D. Brickley and L. Miller, "FOAF Vocabulary Specification," Jan. 2014. `http://xmlns.com/foaf/spec/`.

[32] K.-L. Skillen, L. Chen, and e. a. Nugent, "Ontological user profile modeling for context-aware application personalization," in International Conference on Ubiquitous Computing and Ambient Intelligence, pp. 261–268, Springer, 2012.

[33] M. Sutterer, O. Droegehorn, and K. David, "Upos: User profile ontology with situation-dependent preferences support," in Advances in Computer-Human Interaction, pp. 230–235, IEEE, 2008.

[34] J. Stan, E. Egyed-Zsigmond, A. Joly, and P. Maret, "A user profile ontology for situation-aware social networking," in 3rd Workshop on Artificial Intelligence Techniques for Ambient Intelligence (AITAmI2008), 2008.

[35] G. Klyne, F. Reynolds, C. Woodrow, H. Ohto, J. Hjelm, M. H. Butler, and L. Tran, "Composite capability/preference profiles (cc/pp): Structure and vocabularies," 2004. https://www.w3.org/TR/CCPP-struct-vocab/.

[36] K.-L. Skillen, L. Chen, C. D. Nugent, M. P. Donnelly, and I. Solheim, "A user profile ontology based approach for assisting people with dementia in mobile environments," in 2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 6390–6393, IEEE, 2012.

[37] M. Bermudez-Edo, T. Elsaleh, P. Barnaghi, and K. Taylor, "Iot-lite ontology," 2015. https://www.w3.org/Submission/2015/SUBM-iot-lite-20151126/.

[38] K. Kotis and A. Katasonov, "An IoT-ontology for the Representation of Interconnected, Clustered and Aligned Smart Entities," 2012.

[39] T. Strang and C. Linnhoff-Popien, "A context modeling survey," in Workshop on advanced context modelling, reasoning and management, UbiComp, vol. 4, pp. 34–41, 2004.

[40] N. D. Rodríguez, M. P. Cuéllar, J. Lilius, and M. D. Calvo-Flores, "A survey on ontologies for human behavior recognition," ACM Computing Surveys (CSUR), vol. 46, no. 4, p. 43, 2014.

[41] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, and P. Steggles, "Towards a better understanding of context and context-awareness," in International symposium on handheld and ubiquitous computing, pp. 304–307, Springer, 1999.

[42] A. K. Dey and J. Mankoff, "Designing mediation for context-aware applications," ACM Transactions on Computer-Human Interaction (TOCHI), vol. 12, no. 1, pp. 53–80, 2005.

[43] N. Ryan, J. Pascoe, and D. Morse, "Enhanced reality fieldwork: the context aware archaeological assistant," Bar International Series, vol. 750, pp. 269–274, 1999.

[44] D. Thevenin and J. Coutaz, "Plasticity of user interfaces: Framework and research agenda.," in Interact, vol. 99, pp. 110–117, 1999.

[45] X. H. Wang, D. Q. Zhang, T. Gu, and H. K. Pung, "Ontology based context modeling and reasoning using owl," in Pervasive Computing and Communications Workshops, 2004. Proceedings of the Second IEEE Annual Conference on, pp. 18–22, Ieee, 2004.

[46] J. Kim and K.-Y. Chung, "Ontology-based healthcare context information model to implement ubiquitous environment," Multimedia Tools and Applications, vol. 71, no. 2, pp. 873–888, 2014.

[47] B. Schilit, N. Adams, and R. Want, "Context-aware computing applications," in Mobile Computing Systems and Applications, 1994. Proceedings., Workshop on, pp. 85–90, IEEE, 1994.

[48] D. Ejigu, M. Scuturici, and L. Brunie, "An ontology-based approach to context modeling and reasoning in pervasive computing," in Pervasive Computing and Communications Workshops, 2007. PerCom Workshops' 07. Fifth Annual IEEE International Conference on, pp. 14–19, IEEE, 2007.