COPri v.2 - A Core Ontology for Privacy Requirements

Mohamad Gharib[a], Paolo Giorgini[b], John Mylopoulos[b]

[a] *University of Florence, Italy*
[b] *University of Trento, Italy*

Abstract

Nowadays, most enterprises collect, store, and manage personal information of customers to deliver their services. In such a setting, privacy has emerged as a key concern since companies often neglect or even misuse personal data. In response to multiple massive breaches of personal data, governments around the world have enacted laws and regulations for privacy protection. These laws dictate privacy requirements for any system that acquires and manages personal data. Unfortunately, these requirements are often incomplete and/or inaccurate as many RE practitioners are insufficiently versed with privacy requirements and how are they different from other requirements, such as security. To tackle this problem, we developed a comprehensive ontology for privacy requirements. In particular, the contributions of this work include the derivation of an ontology from a previously conducted systematic literature review, an implementation using an ontology definition tool (Protégé), a demonstration of its coverage through an extensive example on Ambient Assisted Living, and a validation through competency questions. Also, we evaluate the ontology against the common pitfalls for ontologies with the help of some software tools, lexical semantics experts, and privacy and security researchers. The ontology presented herein (COPri v.2) has been enhanced with extensions motivated by the feedback received from privacy and security experts.

Keywords:
Privacy Ontology, Privacy Requirements, Privacy by Design, PbD, Requirements Engineering, Conceptual Modeling

1. Introduction

It is common practice for most companies today to collect, store, and manage personal information to deliver their services. Therefore, privacy has emerged as a key concern since such companies need to protect the privacy of personal information in order to comply with various privacy laws and regulations that many governments have enacted for privacy protection. For example, the General Data Protection Regulation (GDPR) in the European Union [1], the Protection and Electronic Documents Act (PIPEDA) [2] in Canada, the Information Privacy Principles (IPPs) [3] in Australia, the Health Insurance Portability and Accountability Act (HIPAA) [4] and the Financial Services Modernization Act [5] in the United States.

Accordingly, dealing with privacy concerns is a must these days [6]. However, most of such concerns can be tackled if the privacy requirements of the system-to-be were considered and addressed properly during system design [7, 8]. Unfortunately, most requirements engineers are unfamiliar with privacy requirements and how they differ from other requirements, such as security or vanilla quality requirements [9]. Even when requirements engineers have familiarity with privacy concerns, they focus mainly on confidentiality, and overlook important privacy aspects such as unlinkability, unobservability [7].

Privacy has been studied across multiple disciplines including Law [10], Sociology [11, 12], Psychology [13],

Email addresses: mohamad.gharib@unifi.it (Mohamad Gharib), paolo.giorgini@unitn.it (Paolo Giorgini), john.mylopoulos@unitn.it (John Mylopoulos)

and Information Systems [14] to mention a few. Privacy concepts, which we use to talk and reason about privacy have been studied for more than a century, but still remain elusive and vague concepts to grasp [7, 15]. In recent years, there have been numerous attempts to define privacy in terms of more refined concepts such as secrecy, confidentiality, anonymity, pseudonymity, unlinkability, unobservability, control of personal information [15, 16, 17], or on solitude, intimacy, anonymity, and reserve as in Westin, A. F. [11].

Other studies suggest that the notion of risk is also related to privacy as the loss of control over personal information implies risk [18, 19, 20]. Moreover, Awad and Krishnan [21] investigated how transparency can influence privacy. However, there is no consensus on the definitions of many of these concepts nor which of them should be used to analyze privacy [15]. Besides, many of these concepts are overlapping, thereby contributing to the confusion while dealing with privacy [22].

Ontologies have proven to be a key factor for reducing conceptual vagueness and terminological confusion by providing a shared precise understanding of related concepts [23, 24, 25]. In this context, the main objective of this work is to propose, implement, validate and evaluate a well-defined ontology that captures key privacy-related concepts. It is well acknowledged that privacy is a social concept [26] that depends on how others treat an individual's personal information as well as the social context where such information is captured and used [6]. Accordingly, a privacy ontology should conceptualize privacy requirements in their social and organizational setting.

The contributions of this work include the derivation of an ontology from a previously conducted systematic literature review, an implementation using an ontology definition tool (Protégé), a demonstration of its coverage through an extensive example on Ambient Assisted Living, and a validation through competency questions. Also, we evaluate the ontology against the common pitfalls for ontologies with the help of some software tools, lexical semantics experts, and privacy and security researchers.

This paper is an extension of Gharib et al. [27]. The extensions, motivated by the feedback received from privacy and security experts, amount to doubling the content of the paper and have as follows:

- We extend and improve the ontology by refining the *personal information* related concepts, and we also integrate the *consent* concept in the ontology, which allows for a better analysis of privacy requirements. Moreover, we extend and refine the *purpose of use* related concepts allowing more expressive analysis to determine whether the use of *personal information* is *compatible/incompatible* with *consents* provided by *data subjects*.
- We implement the new extensions in the ontology using the Protégé tool[1].
- We extend the analysis support (the Competency Questions (CQs)) to reason about the new extensions.
- We extend the implementation and evaluation of the ontology to account for the new concepts introduced in version 2.

The rest of the paper is organized as follows; Section 2 presents an example concerning an Ambient-Assisted Living (AAL) System that we use to illustrate the applicability and usefulness of our ontology. We describe the process we followed for developing COPri v.2 in Section 3. Section 4 presents the conceptual model of the ontology, and we implement and validate it in Section 5 and Section 6 respectively. We evaluate COPri v.2 in Section 7, and we discuss threats to its validity in Section 8. Related work is presented in Section 9, and we conclude and discuss future work in Section 10.

2. Illustrating example: An Ambient-Assisted Living System

Longevity among the elderly has resulted in many challenges for society and the health care system as well, such as increasing age-related diseases (e.g., Alzheimer's, diabetes, etc.). This has led to a shortage of caregivers [28]. But this is not the only problem since most elderly people (around 89%) prefer to stay at their own homes [28, 29]. Given the costs of home care nursing, it is imperative to develop technologies that help older people live where they feel more comfortable, i.e., at home [29].

[1] `http://protege.stanford.edu/`

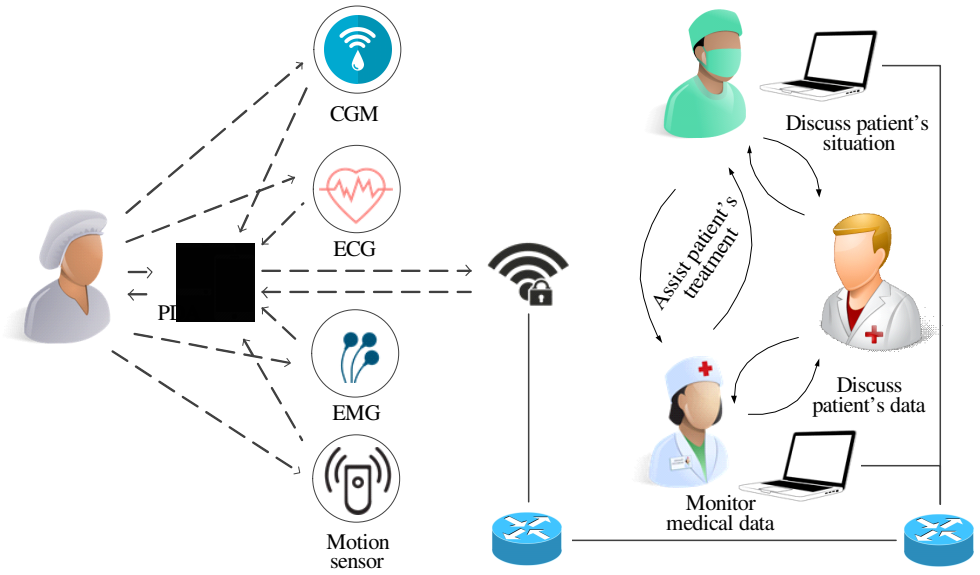

Figure 1: Simplified representation of the AAL system

Ambient-Assisted Living (AAL) systems have been touted as a promising solution to this problem. AAL systems rely on monitoring and actuating devices to shift some of the healthcare services from a hospital-centric to a patient-centric treatment [30]. In other words, instead of being examined face-to-face, a patient's health status can be sensed remotely, continuously, and in real-time. Then, such information is processed and transferred to a health care center [31]. Moreover, AAL technologies facilitate communication among physicians and patients, and allow for discussing medical data and negotiating a treatment procedure remotely [32]. This decreases both the costs of health care services and also the workload of medical practitioners [33, 34, 29]. However, numerous studies showed that privacy is one of the most prominent criticisms for such technologies [35].

Our motivating example concerns an elderly person, Jack, who suffers from diabetes. Jack lives in a home that is equipped with an AAL system that relies on various interconnected body sensors (e.g., Continuous Glucose Monitoring (CGM), location, and motion sensors) to collect data about Jack's vital signs, location, and activities. This information is transmitted to Jack's Personal Digital Assistant (PDA) that assesses his health situation and provides required notifications accordingly. Jack's PDA may also forward such information to a nearby caring center, where a nurse called Sarah can monitor such information, and she can also monitor Jack's activities (e.g., watching TV, sleeping, etc.) by collecting location and motion related-information. Sarah can detect unusual situations and react accordingly, she also has access to all Jack's health records and she may contact the required medical professional that might be needed depending on Jack's situation. Jack, like many other users, wants to preserve his privacy by controlling what is collected and shared concerning his personal information, who is using such information, and for which reasons it is being used. Figure 1 shows a simplified representation of this AAL system.

3. The process for developing the COPri v.2 ontology

The process for developing COPri v.2 (depicted in Figure 2) has been based on [36, 37] following the five principles proposed by Gruber, T. [38] (e.g., clarity, coherence, extendibility, minimal encoding bias, and minimal ontological commitment). The process is composed of five main phases:

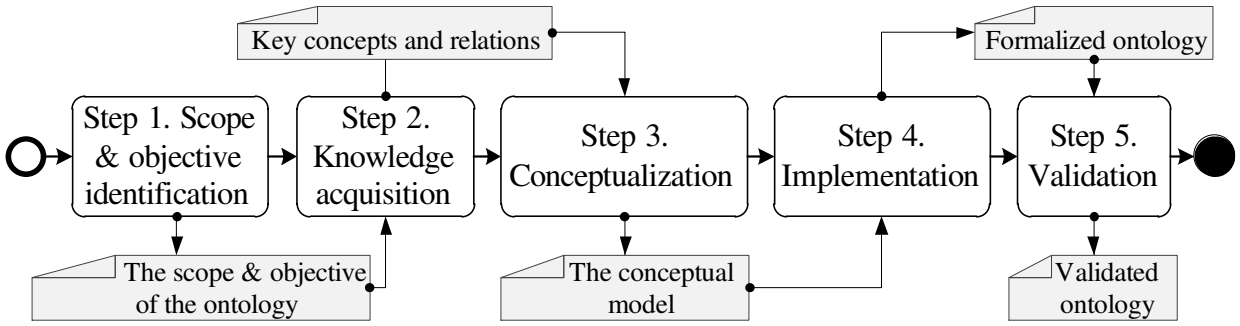

Figure 2: The process for developing the COPri v.2 ontology

- *Step 1. scope & objective identification* aims at identifying the scope of the ontology, the purposes it will be used for, and its intended users [36, 37]. As previously highlighted, there is a need for addressing privacy concerns during system design (e.g., Privacy by Design (PbD) [7, 8]). Nevertheless, based on the results of our systematic literature review [9], most existing studies miss key privacy concepts and relationships. Therefore, it is almost impossible to address main privacy concerns during the system design. To this end, COPri v.2 aims at assisting software engineers while designing privacy-aware systems that belong to various domains by providing a generic and expressive set of key privacy concepts and relationships, which support the elicitation of privacy requirements for the system-to-be in its social and organizational context.

- *Step 2. Knowledge acquisition* aims at identifying and collecting knowledge needed for the construction of the ontology. In [9], we have conducted a systematic literature review for identifying the concepts and relationships used in the literature for capturing privacy requirements as well as the semantic mappings between them[2]. The systematic literature review has identified 38 privacy-related concepts and relationships.

- *Step 3. Conceptualization* aims at deriving an ontology that consists of key concepts and relationships for privacy [37]. In [9], we have proposed a preliminary ontology consisting of 38 concepts and relationships, which has been extended to 52 concepts and relationships in [27]. In this paper, we further extend and refine our earlier proposal to a comprehensive ontology consisting of 63 concepts and relationships. A detailed description of the resulting ontology (COPri v.2) is provided in the next section.

- *Step 4. Implementation* codifies an ontology in a formal language. This requires an environment that guarantees the absence of lexical and syntactic errors from the ontology, and an automated reasoner to detect inconsistencies and redundant knowledge. Although there exist several environments for developing (codifying) ontologies (NeOn Toolkit [40], OntoEdit [41], SWOOP [42], Protégé [43]), we have chosen Protégé[3] that is a set of open-source and domain-independent ontology design software. Protégé can be used easily for creating, modifying, visualizing and checking the consistency of ontology. Moreover, the reasoner can be used to automatically compute a classification hierarchy (*inferred hierarchy*) based on a manually constructed class hierarchy that is called the *asserted hierarchy*. In addition, Protégé offers several useful plug-ins for visualizing ontology, and most importantly it offers a plug-in for using SPARQL (Protocol and RDF Query Language) to extract knowledge from an ontology

[2] A detailed version of the systematic literature review can be found at [39]

[3] `http://protege.stanford.edu/`

through queries and rules [44]. The implementation of COPri v.2 is discussed in section 5.

- *Step 5. Validation,* aims at ensuring that the resulting ontology meets the needs of its usage, in our case, the representation of privacy requirements [37]. According to [36], informal and formal questions/queries can be used to validate ontology. Following [45, 46], we validated COPri v.2 after applying it to the AAL illustrative example and querying the ontology instances depending on Competency Questions (CQs). Then, evaluating whether the ontology captures enough detailed knowledge about the targeted domain to fulfill the needs of its intended use. The validation of COPri v.2 is discussed in more detail in section 6.

4. The COPri v.2 ontology

The ontology is presented as a UML class diagram in Figure 3, where the concepts of the ontology are organized into four main dimensions:
(1) Organizational dimension includes concepts for capturing the social and organizational aspects of the system, which are organized into several categories:

Agentive entities. They capture active entities that are intentional, have goals and carry out actions towards their fulfillment. They include the following six concepts and two relationships:

An actor represents an autonomous entity that has intentionality and strategic goals and can carry out actions towards their fulfillment. The *Actor* concept covers two sub-categories: a *role* and an *agent*, where the *role* concept covers three sub-categories.

A role represents an abstract actor with an associated set of behaviors and capabilities within some specialized context. Moreover, we have defined three roles that represent key entities, who have special behaviors and functionalities related to personal information[4]:

Data Subject (DS) represents an identifiable natural person, who can be identified directly or indirectly by reference to an identifier such as a name, location data, etc. [1, 47, 48, 49].

Data Controller (DC) represents a natural or legal person, public authority, agency or other body which, alone or jointly with others, which determines the purposes and means of the processing of personal information [1, 47, 49, 50].

Data Processor (DP) represents a natural or legal person, public authority, agency or any other body, which processes personal information on behalf of the Data Controller [1, 47, 48, 49, 50].

An agent represents an autonomous entity that has a specific manifestation in the system.

Is_a represents a relationship between two *roles*, where one role is a specialization of the other.

Plays represents a relationship between an *agent* and a *role*, where an *agent* can **play** a *role* or more, in which case it inherits the properties of the role it plays.

Intentional entities. They capture objectives that active entities aim for achieving. They include one concept and two relationships:

A goal is a state of affairs that an actor intends to achieve. When a goal is too coarse to be achieved, it can be refined through *and/or-decompositions* into finer subgoals.

And-decomposition represents a relationship between a *goal* (a parent goal) and at least two other *goals* (subgoals), such that the parent-goal is achieved if all of its sub-goals are achieved.

Or-decomposition represents a relationship between a *goal* (a parent goal) and at least two other *goals* (subgoals), such that the parent-goal is achieved if at least one of its sub-goals is achieved

Informational entities. They capture informational assets (e.g., information, personal information). They include seven concepts and two relationships:

[4]Other roles exist as well, yet we chose to include only the most prominent and commonly used roles to minimize the number of the concepts in the ontology

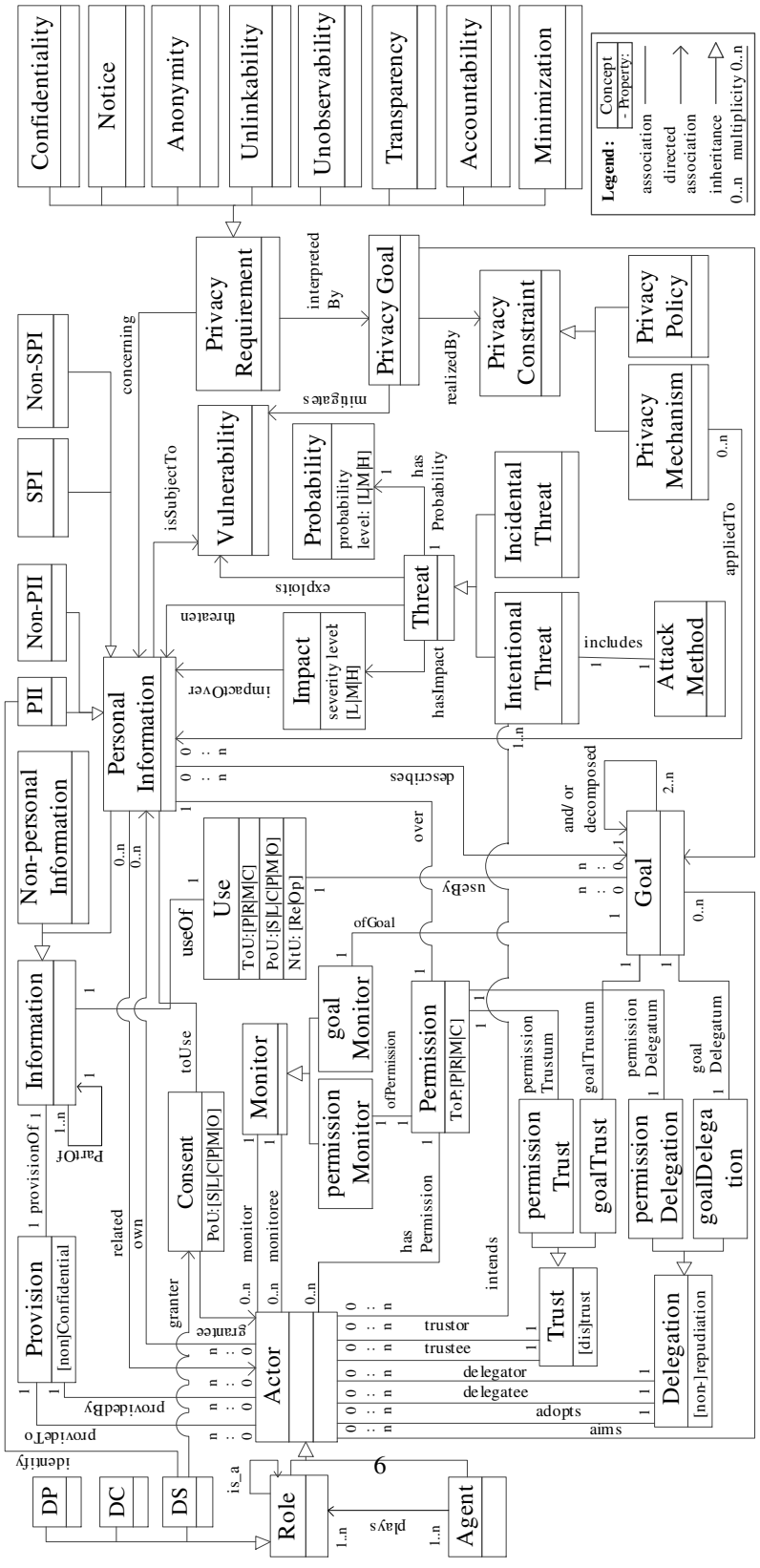

Figure 3: The conceptual model of COPri v.2

Information represents a statement provided or learned about something or someone. We differentiate between two types of information:

Non-Personal information, any information that cannot be *related* (directly or indirectly) to an identified or identifiable legal entity, or personal information that has been made public by its legal owner (Data subject) [8].

Personal information, any information that can be *related* (directly or indirectly) to an identified or identifiable legal entity (e.g., names, addresses, medical records, etc.), who has the right to control how such information can be used by others [51, 52]. *Personal information* can be further specialized into two main types:

Personally Identifiable Information (PII), any information that can be used, on its own, to distinguish, trace and/or *identify* an individual's identity.

Non-Personally Identifiable Information (Non-PII), any personal information that cannot be used, on its own, to distinguish, trace and/or *identify* an individual's identity.

Sensitivity of personal information. Personal information can be specialized into *Sensitive Personal Information (SPI)* and *Non-Sensitive Personal Information (Non-SPI)*, depending on information type (e.g., private, intimate) as well as the *state of affairs* relevant to such information, i. e., when, where and for which purposes such information has been collected. However, determining what *state of affairs* determine sensitivity is beyond the scope of COPri v.2.

PartOf represents a relationship between between an information entity and its sub-parts. In particular, information can be atomic or composite (composed of several parts), and we rely on *partOf* to capture such relationship.

Describes is a relationship where information characterizes a goal (activity) while it is being pursued by some actor. The Ontology has been extended with *Collect* and *Describes* to capture situations when information *describing* some activities performed by a data subject is being *collected* by others.

Information use is a relationship between a goal and information, and it has three attributes:

i- Type of Use (ToU), our ontology provides four types of use: *Produce, Read, Modify,* and *Collect,* indicates that information is created, consumed, altered and acquired respectively.

ii- Need to Use (NtU) captures information relevant to the achievement of a goal, and there are two types of *NtU*: *Require* and *Optional*, wherein the first the use of information is required for the goal achievement, and in the later is not [56].

iii- Purpose of Use (PoU), captures the purpose(s) for which personal information would be used. Following [57], we differentiate between six types of PoU: *1- (S)ervice Purpose*, any purpose related to providing services to individuals, including advertisements, preference-based content, etc. *2- (L)egal Purpose*, any purpose related to complying with court orders, regulatory purposes or any other legal reasons. *3- (C)ommunication Purpose*, any purpose related to communicating individuals about products, services, update about services and/or new products, and other related purposes. *4- (P)rotection Purpose*, any purpose related to information protection, fraud detection, potential misuse identification, etc. *5- (M)erger Purpose*, any purpose related to mergers, transfer of control, or transfer of company/entity that is managing the personal data. *6- (O)ther Purpose*[5], any other purpose that is not covered by the previous purposes.

Ownership, Permission & Consent. They capture who and how can control the use of personal information. They include one relationship and two main concepts:

Owns is a relationship, which indicates that an actor is the legitimate owner of information.

[5] Bhatia and Breaux [57] named this purpose *Vague Purpose*

Permission is an authorization that identifies a particular use of particular information in a system. Information owner (data subject[6]) has full control over the use of information it owns, and it depends on *permissions* for such control. In COPri v.2, a permission has a type that can take as values (P)roduce, (R)ead, (M)odify and (C)ollect, which cover the four relationships between goals and information that our ontology proposes.

Consent represents an agreement at the Data Subject side concerning the purpose of use of information it owns (e.g., personal information) [1, 48, 58]. A *consent* is *granted* by a Data Subject (called *granter*) to another actor (called *grantee*) concerning information it owns for a specific *purpose* of *use*, where such *purpose* has a type that can take as values (S)ervice, (L)egal, (C)ommunication, (P)rotection, (M)erger or (O)ther Purposes.

Entity interactions: capture the interactions/dependencies among actors of the system concerning their objectives and entitlements. The ontology adopts three types of interactions:

Information provision captures the transmission of information (*provisionOf*) by an actor (*provisionBy*) to another one (*provisionTo*), where the source of the provision relationship is the provider and the destination is the requester. Moreover, *Information provision* has a type that can be either *confidential* or *nonConfidential*, where the former guarantee the confidentiality of the transmitted information, while the last does not.

Delegation indicates that actors can delegate obligations and entitlements to one another, where the source of delegation called the delegator, the destination is called delegatee, and the subject of delegation is called delegatum. The concept of *delegation* is further specialized into two concepts: *Goal delegation,* where the delegatum is a goal; and *Permission delegation,* where the delegatum is a permission.

[6]We treat "information owner" and "data subject" as synonyms

Adoption is considered as a key component of social commitment, and it indicates that an actor accepts to take responsibility for the delegated objectives and/or entitlements from another actor [59].

Entity social trust: the need for trust arises when actors depend on one another for goals or permissions since such dependencies might entail risk [60]. Trust captures the actors' expectations in one another concerning their dependencies (e.g., delegated goals/permissions). *Trust* has a type that can be either: *(1) Trust* means the trustor expects that the trustee will behave as expected considering the trustum (e.g., a trustee will not misuse the trustum), and *(2) Distrust* means the trustor expects that the trustee may not behave as expected considering the trustum. Moreover, the concept of *Trust* is further specialized into two concepts *GoalTrust,* where the trustum is a goal; and *PermissionTrust,* where the trustum is a permission.

Monitoring: is the process of observing and analyzing the performance of an actor in order to detect any undesirable performance. We adopt the concept of *monitoring* to compensate for the lack of trust or distrust in the trustee concerning the trustum. The concept of *monitor* is further specialized into two concepts *GoalMonitor,* where the subject of the monitoring is a goal; and *PermissionMonitor,* where the subject of the monitoring is a permission.

(2) Risk dimension includes risk related concepts that might endanger privacy needs at the social and organizational levels:

A vulnerability is a weakness in the current state-of-affairs that may be *exploited* by a *threat*.

A threat is a potential incident that *threatens* personal information by *exploiting* a *vulnerability* concerning such information [61]. In COPri v.2, we differentiate between two types of threat:

(1) Intentional threat is a threat that requires a *threat actor* and *includes* a presumed *attack method* [39]. A **threat actor** is an actor that intends to achieve an *intentional threat* [61], and an **attack method** is a standard means by

which a *threat actor* aims to achieve an *intentional threat* [25, 61]. In particular, an *attack method* is employed by a *threat actor* aiming to achieve an *intentional threat*.

(2) Incidental threat is a casual, natural or accidental threat that is not caused by a *threat actor* nor does it require an *attack method*.

A *threat* has a *probability* that measures the likelihood of its occurrence (i.e., it will become a reality), and it is characterized by three different values *high*, *medium* or *low*. For instance, the probability of an *incidental threat* can be assessed depending on the occurrence likelihood of related natural/accidental causes that leads to such threat. While the probability of an *intentional threat* can be assessed based on the success likelihood of the employed attack method. For example, some vulnerabilities are either not detected or not associated with potential attacks during the vulnerability management activity. While such vulnerabilities remain unknown to attackers, the probability of related threats remains very low. Yet if such vulnerabilities are detected by an attacker and disclosed publicly, the probability of related threats become extremely high since there is almost no defense against attacks that exploit such vulnerability (e.g., zero-day attack).

Impact is the expected consequence of a *threat* over the personal information. An *impact* has a *severity* that captures the level of the impact [25], and takes values *high*, *medium* or *low*.

(3) Treatment dimension includes concepts to mitigate risks:

A privacy goal defines an intention to counter threats and prevent harm to personal information by satisfying privacy properties.

A privacy constraint is a design restriction that is used to realize/satisfy a privacy goal, constraints can be either a privacy policy or privacy mechanism.

A privacy policy defines permitted and forbidden actions to be carried out by actors toward information.

A privacy mechanism is a concrete technique that operationalizes a privacy goal. Some mechanisms can be directly *applied to personal information* (e.g., anonymity, unlinkability).

(4) Privacy dimension includes concepts to capture the data subjects' privacy requirements/needs concerning their personal information:

Privacy requirements capture data subjects' privacy needs. *Privacy requirements* can be *interpretedBy privacy goals*, and it is further specialized into eight more refined concepts[7]:

Confidentiality means personal information should remain inaccessible to incidental or intentional threats [8, 15, 53]. We rely on the following three principles to analyze confidentiality:

1- Non-disclosure, personal information can only be disclosed if the data subject's permission is provided [8, 15, 53]. Therefore, *non-disclosure* can be analyzed depending on the existence of read permission as well as the confidentiality of information provision. Note that *non-disclosure* also covers information transmission that is why we differentiate between two types of information provision (e.g., confidential, non-confidential).

2- Need to Know (NtK) an actor should only use information if it is strictly necessary for completing a certain task [8, 62]. **NtK** can be analyzed depending on *Need to Use (NtU)* that captures the necessity of use, i.e., personal information can only be used if it is strictly necessary for completing a certain task [8].

3- Purpose of Use (PoU), means personal information can only be used for specific and legitimate purposes [15, 53], which are *compatible* with the purposes specified in the *consent* provided by its *data subject*, and not in ways that are incompatible with those purposes.

[7]The right to erasure (right to be forgotten) is essential in several privacy laws, yet we did not consider it since the use of information is limited to a specific and explicit purpose (a goal), i.e., information will not be kept after achieving the goal

Anonymity means personal information should be used without disclosing the identity of its data subject [15, 17, 53]. Personal information can be *anonymized* (e.g., removing identifiers) depending on some *privacy mechanism*. *Anonymity* might be achieved if the primary/secondary identifiers of a data subject (e.g., name, social security number, address, etc.) is removed or substituted.

Unlinkability means that it should not be possible to link personal information back to its data subject [7, 17, 63]. A *privacy mechanism* can be used to remove any linkage between personal information and its data subject. Note that *Anonymity* cannot guarantee *unlinkability*, and each of them does not imply the other [17]. For example, an attacker might link information back to a specific data subject (linkability threat) without necessarily revealing the identity of the data subject (identifiability threat), i.e., the attacker does not know the real identity of the data subject.

Unobservability aims at hiding activities (e.g., use a resource or service) that are performed by a data subject [17, 64, 65]. Unlike *Anonymity* and *Unlinkability* that try to hide the identity of the data subject and any linkage between information and data subject respectively, in *unobservability* it should be impossible to others to know whether a data subject has/has not performed an activity. *Unobservability*[8] can be analyzed relying on the *describes* relationship, which enables for detecting situations where personal information that describes an activity (goal) being pursued by a data subject is being collected by some other actor [66].

Notice means a data subject should be notified when its information is being collected [15, 53]. *Notice* can be analyzed depending on the collect relationship and its corresponding permission. In the case where personal information is being collected and there is no permission to collect it, a notice violation will be raised. Providing a permission to collect implies that the actor has been already notified and agreed upon the collection of his information.

Minimization means the collection of Personally Identifiable Information (PII) should be kept to a strict minimum. Wherever possible, identifiability, observability, and linkability of personal information should be minimized [17]. Since the identifiability, linkability, and observability of personal data are already covered by the *anonymity*, *unlinkability* and *unobservability* requirements respectively, we focus on keeping the collection of PII as minimum as possible. However, it is not always easy to specify whether the collection of PII is strictly required, yet the ontology can identify whether PII is collected and by which goal, leaving such knowledge to be further analyzed by experts to determine whether such collection is strictly required or not.

Transparency means a data subject should be able to know who is using its information and for what purposes [53], we rely on two principles to analyze transparency:

- **Authentication** a mechanism aims at verifying whether actors are who they claim they are, and it can be analyzed by verifying whether i) the actor is playing a role that enables the identification of its main responsibilities; and ii) the actor is not playing any threat actor role.
- **Authorization** a mechanism aimed at verifying whether actors can use information in accordance with their credentials [53].

Accountability means a data subject should be able to hold information users accountable for their actions concerning its information [53]. We rely on the *non-repudiation* principle to analyze accountability, which can be analyzed relying on the adoption relationship, i.e., if a delegatee did not adopt the delegatum, a *non-repudiation* violation can be raised.

Note that several extensions and modifications in CO-Pri v.2 ontology were motivated by the feedback received from privacy and security experts. In particular, we have conducted a survey with privacy and security experts to

[8] Achieving Unobservability implies that Undetectability has been achieved [17]. Therefore, we do not consider Undetectability in our ontology

evaluate the adequacy and completeness of the COPri ontology in terms of its concepts and relationships for dealing with privacy requirements. More specifically, the main objective of the survey was twofold: (a) whether the ontology includes unrequired concepts/relationships; and (b) whether the ontology misses important privacy-related concepts/relationships. The survey will be discussed in details in Section 7.

5. The implementation of COPri v.2

We have implemented the COPri v.2 ontology using the Protégé tool[9] that supports the creation, modification, visualization and consistency-checking for an ontology. Protégé also offers a plug-in for using SPARQL to query an ontology. In particular, we have implemented COPri v.2 relying on classes and object properties (relationships) in Protégé, had to amend and/or create new classes and relationships during this process. For each class that has attributes with quantitative values, we have created a class (called a Value Partition pattern) to present such attributes, and several individuals (instances) to cover all quantitative values of their corresponding attributes. For example, the *Probability Level* attribute of the *Probability* class that its value can be *High*, *Medium* or *Low* , has been represented by a class named *Probability level* that has three defined individuals *plhigh*, *plmedium* and *pllow*. Furthermore, we have defined the *hasProbability* property to link the *Probability* class to the *Probability Level* class.

Classes may overlap and to ensure that an individual that belongs to one of the classes cannot be a member of any other class, such *classes* must be made disjoint from one another. Thus, all *primitive siblings* classes (e.g., Personal Information and Non-Personal Information) in our implementation of the ontology have been made *disjoint*. This helps the reasoner to check the logical consistency of the ontology. Moreover, we have used Probe Classes [67], which are classes that are subclasses of two or more disjoint classes to test and ensure that the ontology does not include inconsistencies. We have also used a covering axiom to solve the open-world assumption in OWL-based ontologies, where a covering axiom is a class that results from the union of the classes being covered. For example, Personal Information and Non-Personal Information are the only subclasses of the Information class, and using a covering axiom here means that Information must be one of these two subclasses, i.e., Information is *covered* by Personal Information and Non-Personal Information.

In the Protégé tool, relationships between classes are called object properties (properties for short), and they are used to link individuals (instances) from a class to individuals from another class. The source of the property is called the domain class and the destination of the property is called the range class. Specifying the domain and range for properties can be used by the reasoner to make inferences and detect inconsistencies in the ontology. In this context, we have defined the domain and range for each of the properties. Table 1 shows the domain and range for each of the properties of the ontology. In which, we can identify the domain of the *aims* property (relationship) is the *Actor* class and its range is the *Goal* class, which means that the *aims* property is supposed to link individuals from the class *Actor* to individuals from the class *Goal*.

Another example, is the *identify* property that takes the class *Personally Identifiable Information (PII)* as a domain and the class *Data Subject* as a range. This allows the reasoner to infer the *Data Subject* for each *PII* while querying the ontology relying on Competency Questions (CQs), which will be discussed in the next section. Additionally, we defined only one inverse property (e.g., the related property between Personal Information and actor classes) in our ontology to minimize the number of object properties. Finally, we have used cardinality restrictions to specify the number of relationships between classes depending on at *least*, at *most* or *exactly* keywords.

A snapshot of the COPri v.2 ontology is shown in Figure 4[10]. In which, we can identify the owl:Thing class. Like other OWL ontologies, all classes of the COPri v.2 ontology are subclasses of the owl:Thing class. The relationships between a class and its subclasses are, usually, represented by light blue arcs. A relationship between a class and its individuals is, usually, represented by a purple arc. For all other relationships between the various classes of the ontology, OntoGraf tries to assign different

[9] `http://protege.stanford.edu/`

[10] The COPri v.2 ontology is available in OWL formal at `https://bit.ly/30TjE70`

Table 1: Description of the domain and range of object properties

Object prop.	Domain	Range	Object prop.	Domain	Range
adopts	Actor	Delegation	aims	Actor	Goal
andDecomposed	Goal	Goal	appliedTo	Pri.Mechanism	Per.Information
concerning	Pri.Requirement	Per.Information	delegatee	Delegation	Actor
delegator	Actor	Delegation	describes	Per.Information	Goal
goalTrustum	Trust	Goal	goalDelegatum	goalDelegation	Goal
granter	DataSubject	Consent	grantee	Consent	Actor
hasConsentPoU	Consent	PoUType	identify	PII	DataSubject
hasDelegationType	Delegation	DelegationType	hasImpact	Threat	Impact
hasNeedtoUseType	Use	NeedtoUseType	hasPermission	Actor	Permission
hasPermissionType	Permission	PermissionType	hasProbability	Threat	Probability
hasProvisionType	Provision	ProvisionType	hasPoUType	Use	PoUType
hasSensitivity	Per.Information	SensitivityLevel	hasSeverityLevel	Impact	SeverityLevel
hasTrustLevel	Trust	TrustLevel	hasTypeOfUse	Use	TypeOfUse
impactOver	Impact	Per.Information	includes	Int.Threat	AttackMethod
intends	Actor	Int.Threat	interpretedBy	Pri.Requirement	PrivacyGoal
is_a	Role	Role	isSubjectTo	Per.Information	Vulnerability
mitigates	PrivacyGoal	Vulnerability	monitor	Actor	Monitor
monitoree	Monitor	Actor	ofGoal	goalMonitor	Goal
ofPermission	perm.Monitor	Permission	orDecomposed	Goal	Goal
over	Permission	Per.Information	own	Actor	Per.Information
partOf	Information	Information	perm.Delegatum	perm.Delegation	Permission
perm.Trustum	Trust	Permission	plays	Agent	Role
provideTo	Provision	Actor	provideBy	Actor	Provision
provisionOf	Provision	Information	realizedBy	PrivacyGoal	Pri.Constraint
related	Per.Information	Actor	threaten	Threat	Per.Information
trustee	Trust	Actor	trustor	Actor	Trust
usedBy	Goal	Use	usedOf	Use	Information
toUse	Consent	Per.Information	exploits	Threat	Vulnerability

colors.

Note that some elements of the implementation do not correspond exactly to the UML class diagram because of discrepancies between the modeling elements of UML and Protégé. For example, since Protégé will consider the four sub-categories of PI (e.g., PII, non-PII, SPI, and non-SPI) as primitive siblings, which is not correct. We have considered PII and non-PII as the only sub-categories of PI, and to represent SPI and non-SPI, we have created a *Sensitivity* class that has a *Sensitivity level* attribute, which can be *Sensitive* (e.g., *slSensitive*) or *Non-Sensitive* (*slNon-Sensitive*). Furthermore, we have defined the *has-Sensitive* property to link the *Personal Information* class to the *Sensitivity level* class.

6. Validation

In this section, we discuss how we validated COPri v.2 depending on Competency Questions (CQs), which represent a set of queries that the ontology must be capable of answering to be considered competent for conceptualizing the domain of discourse [36, 46]. In other words, the ontology contains "all" necessary and relevant knowledge, if it can correctly answer the CQs. In this context, CQs specify what knowledge has to be entailed in the ontology and, thus, can be seen as a set of requirements on the content that has to be represented in the ontology. Since CQs are mainly used to assure that the ontology is competent for conceptualizing the domain of discourse, the CQs have been extended and refined to cover the new knowledge we obtained about the domain of discourse while developing the ontology. In particular, the final list of CQs is an extended and modified version of the initial list of CQs that we have considered when we start developing the ontology.

Moreover, the feedback received from the privacy and security experts helped in better covering the domain of disclosure, leading to extend and refine both the CQs and

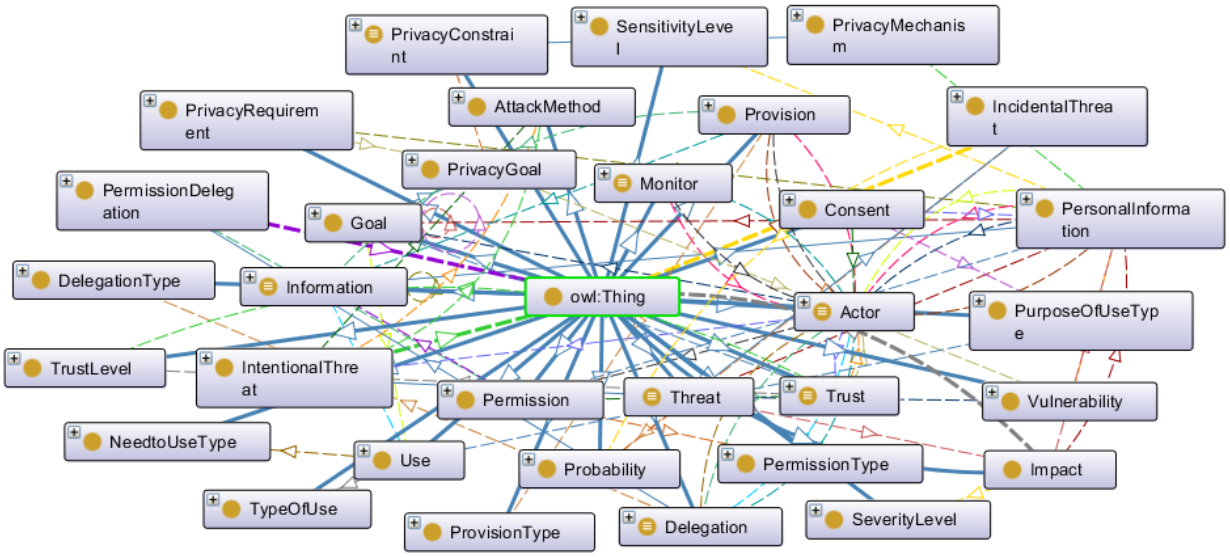

Figure 4: A snapshot of the COPri v.2 ontology using the OntoGraf plug-in

the concepts and relationships of the ontology to overcome some revealed limitations (e.g., the ontology does not cover some aspects of the domain of disclosure) and inadequacies (e.g., the ontology cannot be used to return a desired/correct answer) in the ontology. More specifically, when the ontology does not cover some aspects of the domain of disclosure, we extended its concepts to cover these aspects, which allows refining and extending related CQs. When the ontology cannot correctly answer a CQ, the concepts of the ontology are extended and/or refined in a way that allows returning the desired/correct answer.

For instance, we received a suggestion to refine the agentive entities and their relationships with the information concept. As a response, we have added three new roles (e.g., *DS*, *DC*, and *DP*) and one relationship (e.g., *identify* that links *PII* and *DS*). This allows extending the CQs with CQ4 and refining several existing CQs, namely: CQ18, CQ19, CQ21- CQ26, and CQ28. Another suggestion was the inclusion of the *consent* concept, thus, adding this concept enabled us to formulate a new CQ, namely: CQ21. Moreover, they suggested refining the *Purpose of Use (PoU)* analysis, which motivates formulating a new CQ (e.g., CQ26), and refining another one, namely CQ21. The experts' feedback will be discussed in the next section.

We validated the COPri v.2 ontology by applying it to the AAL illustrating example, and then, query the ontology instance relying on CQs and check whether these queries can return correct answers, i.e., assessing whether the ontology can capture detailed information about the domain of discourse and fulfill the needs of its intended use [37]. In particular, the CQs are meant to assist and guide requirements engineers while dealing with privacy requirements by capturing main wrong/bad design decisions (we call *violations*) related to the four dimensions of our ontology, namely organizational, risk, treatment, and most importantly privacy requirements (e.g., confidentiality violation, notice violation, etc.).

To this end, 29 CQs[11][12] have been defined (shown in Table 2[13]), which we consider sufficient for capturing enough knowledge about the system, bad design decisions

[11]Note that the main focus of the CQs is privacy requirements, not goal analysis

[12]The usability and utility of the CQs are yet to be evaluated with potential end-users

[13]The formalization of the CQs (SPARQL queries) can be found in Appendix A

Table 2: Competency Questions for validating the COPri ontology

	Organizational dimension
CQ1.	Who are delegators that delegate produce | read | modify | collect permission that is not accompanied by trust nor monitoring? - (returns also information and delegatees)
CQ2.	Who are delegators that delegate produce | read | modify | collect permission that is accompanied by both trust and monitoring? - (returns also information and delegatees)
CQ3.	Who are delegatee that can repudiate acceptance of a delegatum? - (returns also delegator)
CQ4.	What are the role(s) each agent is playing?
CQ5.	What personal information is Sensitive/NonSensitive?
	Risk dimension
CQ6.	What are existing vulnerabilities and the personal information each relates to?
CQ7.	What are existing vulnerabilities and the threats that can exploit them?
CQ8.	What are existing vulnerabilities that are not mitigated by privacy goals?
CQ9.	What are existing threats and the personal information each threatens?
CQ10.	What are existing threats that have an impact with severity level Low | Medium | High over personal information?
CQ11.	What are existing intentional threats and the personal information they each threaten?
CQ12.	Who are threat actors and the intentional threats they intend to perform?
CQ13.	What are existing attack methods and the intentional threats they can be used for?
CQ14.	What are existing incidental threats and the personal information they threaten?
CQ15.	What are existing threats of probability Low | Medium | High?
	Treatment dimension
CQ16.	What are privacy goals that are realized by privacy constraints? - (returns also privacy constraints)
CQ17.	What are existing privacy mechanisms and the personal information they are applied to?
	Privacy dimension
CQ18.	What personal information is read without read permissions? - (returns also data subject, misusing actor and the goal using such information)
CQ19.	What personal information is transferred relying on non-confidential provision? - (returns also data subjects)
CQ20.	What personal information is used by a goal, where its usage (*NtU*) is not strictly required (i.e., optional)? - (returns also personal information)
CQ21.	What personal information is used by goals, where purpose of use (*PoU*) is incompatible with consents provided by their data subjects? - (returns also personal information, the using goal, and the misusing actor)
CQ22.	What personal information can disclose the identity of their data subjects (not anonymized)? - (returns also data subjects)
CQ23.	What personal information can be linked back to its data subject? - (returns also data subject)
CQ24.	What personal information describes a goal, and is also collected by an actor? - (returns also goal and data subject)
CQ25.	Who are the actors that collect personal information without collect permissions? - (returns also personal information and data subjects)
CQ26.	What Personally Identifiable Information (PII) is being collected for (S)ervice | (L)egal | (C)ommunication | (P)rotection | (M)erger | (O)ther purpose? - (returns also the using goal and data subjects)
CQ27.	Who are the actors that do not play any role or play a threat role?
CQ28.	Who are the actors that are using (produce, read, modify or collect) personal information without required permission? - (returns also information, data subjects and the using goal)
CQ29.	Who are delegatees that have not adopted their delegatum? - (returns also delegatum and delegator)

and violations to the privacy requirements considered in our ontology. In what follows, we describe each of these four groups of CQs:

CQ1-5 are dedicated to query organizational aspects,

where *CQ1* can be used to capture situations where a permission is delegated without a trust or trust compensation (e.g., monitoring). With the absence of trust and monitoring relationships, the delegator cannot guarantee that the delegatee will not misuse the delegated permission. Considering our illustrating example, if there was no trust nor monitoring between Jack and Sarah concerning the delegation of read and/or collect permissions of Jack's location, *CQ1* will detect and report such a situation. *CQ2* can be used to capture situations, where an actor monitors a delegation of permission although he/she trusts the delegatee. In such a situation, monitoring is not required and it is considered as a bad design decision. Concerning the previous example, if there is also monitoring concerning the delegation of read/collect between Jack and Sarah, *CQ2* will detect such a situation and report that the monitoring relationship is not required.

CQ3 can be used to capture any actor that can repudiate that he/she accepted a delegatum (e.g., a goal or permission), i.e., a delegation without a non-repudiation constraint. *CQ4* can be used to return all agents of the system along with the role(s) that such agents are playing. *CQ5* can be used to return different sets of personal information based on their sensitivity levels (e.g., Sensitive or Non-Sensitive).

CQ6-15 are dedicated to query risk aspects, where *CQ6* can be used to return vulnerabilities (e.g., "V1. Weak masking technique") as well as information that is subject to them (e.g., "I1. Jack's glucose level"). *CQ7* can be used to return vulnerabilities (e.g., "V1.") and threats (e.g., "T1. Linking "I1." information back to Jack") that can exploit such vulnerabilities. *CQ8* can be used to return unmitigated vulnerabilities. *CQ9* can be used to return any threat (e.g., "T1.") that is threatening personal information (e.g., "I1."). *CQ10* can be used to return different sets of threats based on their severity levels (e.g., Low, Medium, or High). *CQ11* can be used to return intentional threats (e.g., "T1.") as well as the personal information (e.g., "I1.") threatened by them.

CQ12 can be used to return threat actors (e.g., an attacker) and the intentional threats they intend for (e.g., "T1."). *CQ13* can be used to return attack methods (e.g., "AM1. De-masking technique") and the intentional threats they are used for (e.g., "T1."). *CQ14* can be used to return incidental threats (e.g., "T2. Leaking the identity of "I1." info owner (Jack)") and personal information that is threatened by them (e.g., "I1."). Moreover, *CQ15* can be used to return different sets of threats based on their probability levels (e.g., Low, Medium, or High).

CQ16-17 are dedicated to query treatment aspects, where *CQ16* can be used to return privacy goals (e.g., "PG1. Ensure anonymity") that have been realized by privacy constraints (e.g., "PC1. Anonymization mechanism"). While *CQ17* can be used to return privacy mechanisms (e.g., "PM1. Remove any linkage between personal information ("I1.") and its owner (Jack)" as well as the personal information (e.g., "I1.") that such mechanisms are applied to.

CQ18-29 are dedicated to query privacy requirements related violations. In particular, *CQ18-21* are used for analyzing *Confidentiality*, where *CQ18-19* are used for analyzing non-disclosure by detecting and reporting when personal information is read without the owner's permission (*CQ18*), or it has been transferred relying on non-confidential transmission means (*CQ19*). *CQ20* is used for analyzing Need to Know (NtK) principle by verifying whether personal information is strictly required by goals using them, i.e., if the Need to Use (NtU) of the goal is optional, *CQ20* will report such violation. *CQ21* is used for analyzing the Purpose of Use (PoU) principle by verifying whether the use of personal information is compatible with the consent provided by its owner, i.e., if the PoU is incompatible, *CQ21* will detect and report such violation.

CQ22 is used for analyzing *Anonymity* by verifying whether the identity of the information owner can be sufficiently identified. For example, if "I1." has not been anonymized relying on some anonymization technique (e.g., "PC1."), *CQ22* will detect and report such violation. *CQ23* is used for analyzing *Unlinkability* by verifying whether it is possible to link personal information back to its owner. For example, if an unlinkability mechanism (e.g., "PC2. Unlinkability mechanism") was not applied to "I1.", *CQ23* will detect and report this violation. *CQ24* is used for analyzing *Unobservability* by verifying whether the identity of the information owner can be observed by others while performing some activity. Consider for example that Jack does not want his activities to be monitored while he is in the bathroom. Then "Jack's location" should not be collected when he is in the bathroom since such information can be used to infer activities that Jack does not want it to be observed by others. If "Jack's location" is collected, *CQ24* will be able to detect

and report such violation.

CQ25 is used for analyzing the *Notice* requirement by verifying whether personal information is being collected without notifying its owner. We consider that providing permission to collect implies that the actor has been already notified and agreed upon the collection of its personal information. In case, personal information is being collected and there is no permission to collect, *CQ25* will detect and report such violation. *CQ26.* is used for analyzing the *Minimization* requirement by focusing mainly on identifying whether PII is collected and by which goal it is collected, which allows experts to determine whether such collection of PII is strictly required or not.

CQ27-C28 are used for analyzing *Transparency*, where *CQ27* analyze the authentication principle and *CQ28* analyze the authorization principle. In particular, *CQ28* verifies whether an actor can be authenticated by checking if it is playing at least one role that enables for identifying its main responsibilities[14], and the actor is not playing any threat actor role. Accordingly, *CQ27* will be able to detect and report whether an actor can be authenticated. While *CQ28* analyze authorization by verifying that actors are not using personal information without the required permissions. In case, Sarah was reading/collecting any of Jack's personal information without a read/collect permission, *CQ28* will be able to detect and report such violation.

Finally, *CQ29* is used for analyzing *Accountability* by verifying whether an actor accepted a delegation, which can be done depending on the adoption concept, if there exists a delegatee without an adopt relationship to the delegatum, *CQ29* will detect and report such violation. Concerning our example, if Sarah did not adopt the read or collect permissions that have been delegated by Jack, *CQ29* will detect and report such violations.

7. Evaluation

We evaluated the COPri v.2 ontology against the common pitfalls for ontologies identified in [68], where the authors classify 20 of these pitfalls under categories 1- *Consistency pitfalls* that check for inconsistencies; 2- *Completeness pitfalls* that check for missing elements; and 3- *Conciseness pitfalls* that check for irrelevant or redundant elements. The pitfalls classification by criteria is shown in Table 3, where we also list the four different methods used for the evaluation. These methods complement each other for the evaluation of the ontology concerning the considered pitfalls. In particular, the first two methods mainly focus on the evaluation of the technical aspects of the ontology (e.g., inconsistencies, cycles in the ontology, misusing OWL primitives). While the third method focuses on evaluating the lexical semantics of the ontology, and the final method evaluates its completeness for dealing with privacy requirements. Note that the evaluation of the ontology with the lexical semantics experts, and the privacy and security experts was performed on the earlier version of the ontology [27], i.e., we did not reevaluate the ontology with them after implementing the modification motivated by their suggestions and feedback.

1- Protégé & HermiT Reasoner[15]: HermiT is one of the first publicly available OWL reasoner, and can perform automated checks for consistency, satisfiability, etc. of OWL-based ontologies. We have used both Protégé & HermiT to perform such checks. More specifically, we used HermiT to detect cycles in the hierarchy of the ontology (*P6.*), and we used OntoGraf plug-in for visualizing the ontology to verify that the ontology does not contain any unconnected elements (*P4.*) Concerning *P10. Missing disjointness*, we made all *primitive siblings* classes *disjoint*, i.e., no missing disjoint can be found in the ontology. Also, we have manually checked whether the domain and range of all object properties have been defined (*P11.Missing domain or range in properties*).

We verified *P14.* and *P16.* depending on Probe Classes[67], which can be used to test and ensure that the ontology does not include inconsistencies. COPri v.2 ontology cannot suffer from *P15.* since we did not use complement operators to describe/define any of the classes, i.e., all defined classes have been defined depending on both necessary and sufficient conditions. The concepts of the ontology are general enough to avoid both *P17. Specializing too much a hierarchy* and *P18. Specifying too much the domain or the range*. No miscellaneous class has been identified (*P21.*), since

[14] If an actor is not playing any role, it will be impossible to authenticate it

[15] `http://www.hermit-reasoner.com/`

Table 3: Pitfalls classification by criteria and how they were evaluated

			Protégé	OOPS!	Experts	Researchers
Consistency	**P1.**	Creating polysemous elements	-	-	✓	-
	P5.	Defining wrong inverse relationships	-	✓	-	-
	P6.	Including cycles in the hierarchy	✓	✓	-	-
	P7.	Merging different concepts in the same class	-	✓	✓	-
	P14.	Misusing "allValuesFrom"	✓	-	-	-
	P15.	Misusing "not some" and "some not"	✓	-	-	-
	P18.	Specifying too much the domain or the range	✓	-	-	-
	P19.	Swapping intersection and union	-	✓	-	-
	P24.	Using recursive definition	-	✓	✓	-
Completeness	**P4.**	Creating unconnected ontology elements	✓	✓	-	-
	P9.	Missing basic information	-	-	-	✓
	P10.	Missing disjointness	✓	✓	-	-
	P11.	Missing domain or range in properties	✓	✓	-	-
	P12.	Missing equivalent properties	-	✓	-	-
	P13.	Missing inverse relationships	-	✓	-	-
	P16.	Misusing primitive and defined classes	✓	-	-	-
Conciseness	**P2.**	Creating synonyms as classes	-	✓	✓	-
	P3.	Creating the relationship "is" instead of using "subclassOf", "instanceOf" or "sameIndividual"	-	✓	-	-
	P17.	Specializing too much a hierarchy	-	-	✓	-
	P21.	Using a miscellaneous class	✓	✓	✓	-

the names of all classes and their sub-classes have been carefully chosen.

2- Evaluation with OntOlogy Pitfall Scanner (OOPS!): OOPS! is a web-based ontology evaluation tool[16] for detecting common pitfalls in ontologies [69]. The COPri v.2 ontology was uploaded to the OOPS! pitfall scanner, which returned an evaluation report[17], where each pitfall is described by its identifier, title, description, elements affected (e.g., classes, object properties, or even the whole ontology) and an importance level. There are three levels of importance based on the impact that a pitfall may have on the ontology: 1- *Critical:* it is crucial to correct the pitfall. Otherwise, the consistency, reasoning, applicability, etc. of the ontology could be affected; 2- *Important:* it is not critical for the functionality of the ontology, but it is important to be corrected; and 3- *Minor:* it does not represent a problem, but correcting it makes the ontology better organized and user friendly.

In summary, OOPS! did not identify any pitfall of types *P2.*, *P4-7.*, *P10-12.*, *P19.*, *P21.*, or *P24.* While it identi-

[16] `http://oops.linkeddata.es/index.jsp`

[17] Evaluation with OOPS! has been performed after evaluating the ontology with Protégé & HermiT, i.e., several pitfalls have been already detected and corrected

fies only one critical pitfall (*P3.*) has been identified stating that we are using is_a relationship instead of using OWL primitives for representing the subclass relationship (rdfs:subClassOf). Yet, is_a relationship is used in most Goal-based modeling languages, where we have adopted many of the concepts and relationships of the COPri v.2 ontology. Therefore, we chose not to replace the is_a relationship with the subClassOf relationship. Moreover, 59 minor pitfalls (*P13.*) have been identified. However, as mentioned earlier we defined very few inverse properties to minimize the number of properties/relationships in the ontology. Finally, two suggestions have been returned, proposing to characterize both is_a and partOf relationships as symmetric or transitive. We took these suggestions into account, characterizing both of these relationships as transitive.

3- Lexical semantics experts: Two lexical semantics experts with a main focus on Natural Language Processing (NLP) have been provided with the COPri ontology, and they were asked to check whether the ontology suffers from any of the following pitfalls[18]: *P1.* Creating polysemous elements, *P2.* Creating synonyms as classes, *P7.* Merging different concepts in the same class, *P17.* Specializing too much a hierarchy, *P21.* Using a miscellaneous class, and *P24.* Using recursive definition. In what follows, we will list some of the issues that have been raised by the experts and how we have addressed them[19].

Several issues have been raised by the experts concerning *P2. Creating synonyms as classes*. For example, one of the experts stated that *"The term "intends" that is used in the definition of the goal concept might be confused with the term "intends" that is used in the definition of the threat actor"*. Therefore, we redefined the definition of the goal concept replacing the term "intends" with the term "aims" as follows: "a goal is a state of affairs that an actor aims to achieve".

One of the experts raised an important issue concerning *P21. Using a miscellaneous class,* he stated *"Dividing information into public information and personal information is not correct (the properties public and personal are not disjoint). I think that the sub-classes should be public information and private information"*. To address this comment, we modified our ontology renaming the two sub-classes covered by the information concept to *Personal Information* and *Non-Personal Information*.

Concerning *P24. Using recursive definition,* we received a comment stating that *"Information is used in its own definition"*. Thus, we redefined Information as "it is a statement provided or learned about something or someone", instead of the old definition "Information represents any informational entity without intentionality". Another comment concerning *P24* was *"Information provision is used in its own definition"*. To tackle this issue, we redefined the Information provision concept replacing the term "provision" with "transmission" as follows: "*Information provision* captures the transmission of information .. ".

4- A survey with researchers: The main purpose of this survey was evaluating the adequacy and completeness of the COPri ontology in terms of its concepts and relationships for dealing with privacy requirements in their social and organizational context (*P9.*). The survey was closed, i.e., it was accessible through a special link that is provided to the invited participants only to avoid unintended participants. In total 25 potential participants from the Requirements Engineering domain with experience in privacy and/or security were contacted to complete the survey, and they were asked to forward the email to anyone who fits in the participating criteria (e.g., has good experience in privacy and/or security requirements). We have received 16 responses (64% response rate).

Survey template design: the survey template[20], and it is composed of four main sections: *S1. General information about the survey* includes a description of the purpose of the survey, privacy and confidentiality statement, and informed consent to be read and accepted (checked) by participants before providing any input. *S2. Participant demographic* includes four questions related to the participant's name, occupation, type of experience (academic and/or industry), and years of experience with privacy and/or security. *S3. Evaluating the COPri ontology* aims at collecting feedback from participates for evaluating the adequacy and completeness of the COPri ontology

[18]The experts' evaluation template can be found at `https://goo.gl/ZEhLnN`

[19]The full experts' feedback and how it was addressed can be found at `https://bit.ly/2GF4UBV`

[20]The survey template can be found at `https://goo.gl/bro8nG`

in terms of its main concept and relationship categories and dimensions. *S4. Final remarks*] aims at collecting suggestions and/or criticisms concerning the COPri ontology.

S2. Result of demographic questions: 15 (93.8%) of the participants are researchers and 1 (6.2%) is a student. Concerning experience with privacy and/or security: 2 (12.5%) of the participants have both academic and industrial experience, and 14 (87.5%) have pure academic experience. Moreover, 3 (18.8%) have less than one year, 7 (43.8%) have between one and four years, and 6 (37.5%) have more than four years of experience.

S3. Result of evaluation questions: this section is composed of 10 subsections, each of them is dedicated to collect feedback concerning the adequacy and completeness of a specific dimension/category of concepts and relationships. In each of these subsections, we provide the definitions of the concepts and relationships of the targeted dimension/category as well as a diagram representing them. Followed by a mandatory question, asking the participant to grade the completeness of the presented concepts and relationships concerning system aspects they aim to capture on a scale from 1 (incomplete) to 5 (incomplete). The result of the evaluation for each of these sections is summarized in Table 4. The result tends to demonstrate that most of the targeted dimensions/categories of concepts and relationships are properly covering the aspects they aim to represent.

Additionally, we have added an optional question in each of the 10 sections to evaluate the adequacy of the concepts and relationships by collecting suggestions to improve the category/dimension under evaluation. Some feedback suggested to refine, include or exclude some of the concepts/relationships, which we will discuss in the following section.

S4. Result of remark questions: most of the feedback was valuable, has raised important issues and ranged from complementing to criticizing. For example, among the encouraging feedback, we received *"COPri covers a wide range of privacy-related concepts, with actor and goal-oriented perspectives, which looks promising. We look forward to seeing it used to capture real-world privacy problem context"*. Another feedback and suggestion was *"I think it is very precise and very good work. Maybe some other concepts could be expressed somewhere"*. Also, we received criticisms such as the following one *"I have no idea how good it is unless it is applied to many real cases. I'm concerned that it is not grounded in reality. It's also very complicated, which makes it hard to apply in the industry"*. However, such criticism opens the way for future research directions.

One the other hand, we received suggestions and comments concerning the refinement of some concepts/relationships[21]. For instance, we received the following comment concerning the agentive entities in the ontology: *"I am not totally sure what is the purpose of representing actors. Are they information subjects?"*. Therefore, we extend the concepts of the agentive entities to include three more roles that represent entities, who have special behaviors and functionalities related to personal information, namely: *Data Subject (DS)*, *Data Controller (DC)* and *Data Processor (DP)*. Moreover, we defined the *identify* relationship between the *Personally Identifiable Information (PII)* and the *DS* concepts to refine the related analysis.

Another comment suggested the inclusion of the consent concept, *"in the GDPR there is the concept of consent, which may be mapped on a privacy requirement"*. That is why we chose to include the concept of *consent* in our ontology, which represents an agreement at the *DS* side concerning the purpose of use of its *personally information*.

Concerning the compatibility of the *Purpose of Use (PoU)* of *personal information*, which was considered in COPri as a property of the *Use* concept and has two values: *Compatible* and *Incompatible*, we have received the following two comments: *"I am not sure that Compliance of Use is a first-class concept? shouldn't it be the result of analysis sometimes?"* and *"I don't think it is appropriate to consider the purpose of use as simply compliant/not compliant"*. To tackle this issue, we replaced the *Compatible* and *Incompatible* values of the PoU with six new values that represent the various purposes, which information can be used for, namely: *(S)ervice*, *(L)egal*, *(C)ommunication*, *(P)rotection*, *(M)erger* or *(O)ther* purposes. Additionally, we extended the *consent* concept with a *PoU* property that has the same six *PoU*. This al-

[21] In COPri [27], we were not able to address all the received feedback due to time restriction. While we tried to address all the raised concerns in this version of the ontology (e.g., COPri v.2)

Table 4: The result of the evaluation

	Strongly disagree	Disagree	N. agree/ n. disagree	Agree	Strongly agree
Q1. Agentive cat.	0 (0%)	1 (6.3%)	3 (18.8%)	6 (37.5%)	6 (37.5%)
Q2. Intentional cat.	0 (0%)	1 (6.3%)	4 (25.0%)	7 (43.8%)	4 (25.0%)
Q3. Informational cat.	0 (0%)	2 (12.5%)	4 (25.0%)	4 (25.0%)	6 (37.5%)
Q4. Goals & info cat.	0 (0%)	2 (12.5%)	2 (12.5%)	6 (37.5%)	6 (37.5%)
Q5. Ownership cat.	0 (0%)	1 (6.3%)	1 (6.3%)	5 (31.3%)	9 (56.3%)
Q6. Interactions cat.	0 (0%)	1 (6.3%)	1 (6.3%)	6 (37.5%)	8 (50.0%)
Q7. Social Trust cat.	0 (0%)	0 (0.0%)	4 (25.0%)	7 (43.8%)	5 (31.3%)
Q8. Risk dim.	0 (0%)	3 (18.8%)	0 (0.0%)	8 (50.0%)	5 (31.3%)
Q9. Treatment dim.	0 (0%)	0 (0.0%)	3 (18.8%)	7 (43.8%)	6 (37.5%)
Q10. Privacy dim.	0 (0%)	2 (12.5%)	2 (12.5%)	5 (31.3%)	7 (43.8%)

lows verifying whether the use of personal information is *Compatible*/*Incompatible* with the type of the *consent* provided by the *DS*.

Finally, we have received the following two criticisms about not considering an operational concept (e.g., a task, a process, etc.) to represent actions that can be performed by agentive entities: *"if actors are agentive entities there should be an entity somewhere representing the actions they can perform"* and *"Can the goal be decomposed into operations an actor can perform?"*. Following several recent goal-based approaches (e.g., [70, 71]) that omitted the use of an operational concept as leaf goals are fine enough to be operationalized, we chose not to include an operational concept to simplify our ontology.

8. Threats to validity

After presenting and discussing our ontology, we list and discuss the threats to its validity in this section. Following Runeson et al. [72], we classify the identified threats under: 1- Internal validity: is concerned with factors that have not been considered in the study, but they could have influenced the investigated factors [72, 73], 2- External validity: is concerned with the degree to which the results of the evaluation can be generalized [72], and 3- Conclusion validity: is concerned with the degree to which conclusions we reached are reasonable/reliable.

1- Internal validity: we have identified two threats: i. *Authors' background,* the authors have good experience in goal modeling (especially in *i** [74] based languages). This may have influenced the selection and definitions of the concepts and relationships of the ontology. However, *i** based languages have been developed with the main objective of capturing requirements in their social and organizational context, which is also a main objective of our ontology. ii. *Concepts inclusion,* the inclusion or exclusion of concepts in our ontology may be subjective as we have favored the inclusion of high-level privacy-related concepts over low-level/fine-grained ones. We did that mainly because we are proposing ontology for requirements engineering, where high-level concepts fit better, and also because most of the fine-grained concepts are, usually, covered by high-level ones. However, to mitigate the threat of excluding key concepts, we have evaluated the completeness of the ontology through a survey that targeted privacy and security experts.

2- External validity: we have identified two threats: i. *Validity of the survey result,* the number of participants can raise concerns about the validity of the result. However, most of them are experts with good experience in privacy, and some of them are high-profile researchers. ii. *Extensive evaluation,* the ontology has been evaluated against the common pitfalls in ontologies with the help of some tools, lexical semantics experts, and privacy researchers, yet it has not been applied in industry, which may reveal undetected errors and new ways to improve it.

However, applying our ontology to real case studies from different domains is on our list for future work.

3- Conclusion validity: although this ontology has been constructed based on concepts obtained through a systematic literature review [9], where the list of these concepts has been further extended and refined in [27]. Then, the completeness of the resulting list has been evaluated by privacy and security experts. We cannot claim that our ontology is complete since it is almost impossible to capture all privacy-related concepts, yet we consider it "sufficiently complete" in that it was designed to subsume all relevant ontologies identified in the systematic literature review.

9. Related work

Several ontologies have been proposed for dealing with privacy and security. For example, Palmirani et al. [75] proposed PrOnto, a first draft privacy ontology that has been developed based on the GDPR. PrOnto aims to support researchers while analyzing privacy policies through SPARQL queries. Unlike COPri, PrOnto mainly focused on the legal aspects of privacy and it does not consider several key concepts such as trust and dependency/delegation that can be used for capturing privacy in its social and organizational settings. Oltramari et al. [76] propose PrivOnto, a semantic framework to analyze privacy policies that rely on an ontology developed to represent privacy-related issues, which can be used by users and/or legal experts for understanding and interpreting them. The authors also developed an interactive online tool that allows users to explore 23,000 annotated data practices instantiated in the PrivOnto knowledge base. However, the main focus of PrivOnto is representing annotated privacy policies, and the concepts (e. g., policy, annotation, practice categories, textual object, etc.) considered in PrivOnto mainly serve this purpose.

Moreover, Singhal and Wijesekera [77] provide a security ontology that can be used to identify which threats endanger which assets and what countermeasures can be used. Although this ontology considered several key privacy-related concepts, they were mainly interpreted in the security general context. Massacci et al. [78] propose ontology for security requirements engineering that adopts concepts from Secure Tropos methodology [63], and several industrial case studies. The ontology captures security requirements in their social and organizational setting, yet it did not consider privacy requirements. Velasco et al. [79] introduce an ontology-based framework for representing and reusing security requirements based on risk analysis. The ontology considers several privacy-related concepts, but its main focus is security and risk analysis.

Additionally, Kang and Liang [80] developed a security ontology for software development that includes most common security concerns, where privacy was considered as a concern. Dritsas et al. [53] developed an ontology for designing and developing a set of security and privacy patterns that can be used to deal with security and privacy requirements for e-health applications. The ontology covers key concepts for dealing with privacy such as stakeholder, security pattern, asset, threat, attacker, countermeasure, etc. However, the ontology mainly focuses on security, thus, it misses key privacy-related concepts. General privacy ontologies/taxonomies (e.g., Anton and Earp [81], Solove et al. [15], and Wuyts et al. [82]) can serve as a general knowledge repository for a knowledge-based privacy goal refinement.

On the other hand, several approaches for dealing with privacy requirements have been proposed in the literature. For instance, Hong et al. [83] propose a privacy risk model specifically for ubiquitous computing, which captures privacy concerns at a high abstraction level, and then refining them into concrete specific solutions. Jensen et al. [84] developed the STRAP method (STRuctured Analysis of Privacy) with a main objective of eliciting and analyzing privacy requirements during system design by representing such requirements as vulnerabilities that need to be addressed. Dritsas et al. [53] developed an ontology for developing a set of security patterns that can be used to deal with security requirements for e-health applications. Besides, Kalloniatis et al. [7] introduce PriS, a security requirements engineering method that considers users' privacy requirements as business goals and provides a methodological approach for analyzing their effect on the organizational processes. Spiekermann and Cranor [85] propose guidelines for building privacy-friendly systems and a three-layer model of user privacy concerns and relate them to system operations in terms of data transfer, storage, and processing. Also, they propose guidelines for building privacy-friendly systems.

Deng et al. [86] provide a methodology for modeling

privacy-specific threats for software systems along with a catalog that can be used to address such threats. Radics et al. [87] introduce the PREprocess, a framework for privacy requirements that aims at guiding software engineers during the elicitation of privacy requirements through the identification of privacy-related patterns. Labda et al. [8] propose a privacy-aware Business Processes framework for modeling, reasoning and enforcing privacy constraints. Moreover, Gharib et al. [6] propose a holistic approach that aims at assisting software engineers in designing privacy-aware systems by providing guidance while dealing with privacy requirements. Finally, Caramujo et al. [88] develop RSL-IL4Privacy that is a domain-specific language for the specification of privacy policies.

In summary, most existing works cover a subset of the four concept categories that we consider in this work, especially, the privacy requirements category. In particular, some works consider a limited number of privacy requirements or confuse privacy requirements with security ones. Other works consider privacy treatment/countermeasure concepts (e.g., pseudonymity) and/or privacy breaches/attacks concepts (e.g., identifiability, disclosure) as privacy requirements. We have avoided such pitfalls while conducting our systematic literature review [9]. This allowed us to carefully select the eight privacy requirements included in this paper, which have been chosen based on two main criteria: (1) their importance for capturing privacy needs, and (2) the frequency of their appearance in various privacy studies. Additionally, these requirements were considered at an appropriate level of abstraction to avoid the selection of too fine-grained concepts that may overlap in meaning and confuse requirements engineers while dealing with them.

10. Conclusions and Future Work

In this paper, we proposed the COPri v.2 ontology for privacy requirements, and since it is based on a systematic literature review; it is more comprehensive in coverage than all ontologies included in our systematic review. Moreover, we implemented the ontology and have applied it to an Ambient-Assisted Living (AAL) systems illustrative example. We have also validated the ontology depending on Competency Questions (CQs), which allows evaluating whether the ontology can capture detailed knowledge about the targeted domain to fulfill the needs of its intended use. Finally, we have evaluated the ontology against common pitfalls for ontologies with the help of several software tools, a lexical semantics expert, and privacy and security researchers.

The main purpose of developing COPri v.2 is assisting requirements engineers while dealing with privacy requirements for systems that handle personal data by providing a comprehensive set of necessary and sufficient concepts that allow for analyzing privacy requirements in their social and organizational context. This work is our second step towards proposing a well-defined privacy ontology, which when completed would constitute a great step forward in improving the quality of privacy-aware systems. However, much work is still to be done.

In this paper, we provide a validity check for the comprehensiveness of our proposal, which needs to be complemented in the future with empirical validation through controlled studies. The next step in this work is to develop a tool and a systematic methodology for privacy requirements that are founded on the COPri v.2 ontology. In particular, an OWL ontology is also a valid Resource Description Framework (RDF) graph (i.e., a set of RDF triples), which can be queried relying on SPARQL queries. Therefore, we aim to assess to which extent the proposed ontology and CQs can be used to capture violations related to privacy requirements (e.g., confidentiality, notice, minimization, linkability violations, etc.) in Linked Data. This will also allow evaluating the usability and utility of the CQs with potential end-users (e.g., requirements engineers).

Additionally, we are planning to develop a goal-oriented framework based on our ontology. This framework will be used for modeling and analyzing privacy requirements in their social and organizational context[22]. Moreover, it will provide mechanisms for deriving the final privacy specifications in terms of privacy policies. This requires achieving two goals, defining privacy policy specification language and a set of rules for the automated derivation of privacy policy specifications from the requirements model. Finally, we aim to promote the adoption of our ontology by providing illustration and documentation as it is available only as a raw OWL file. This

[22] A preliminary version of the extended goal model language can be found in [89]

may encourage other researchers to adopt, use and extend or provide us with useful feedback.

References

[1] E. Parliament, Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Da, Official Journal of the European Communities 59 (2016) 1–88.

[2] Minister of Justice - Government Of Canada, Personal Information Protection and Electronic Documents Act, Tech. rep. (2018).

[3] Office of the Australian information commissioner, Australia. Privacy Act 1988, Tech. rep. (1988).

[4] S. C. f. M. &. Medicaid, The health insurance portability and accountability act of 1996 (HIPAA), Online at URL http://www.cms.hhs.gov/hipaa 25 (1).

[5] Federal Trade Commission, Gramm-Leach-Bliley Act: Financial Privacy and Pretexting, Tech. rep. (2002).

[6] M. Gharib, M. Salnitri, E. Paja, P. Giorgini, H. Mouratidis, M. Pavlidis, J. F. Ruiz, S. Fernandez, A. D. Siria, Privacy Requirements: Findings and Lessons Learned in Developing a Privacy Platform, in: the 24th International Requirements Engineering Conference, RE 2016, IEEE, 2016, pp. 256–265. doi:10.1109/RE.2016.13.

[7] C. Kalloniatis, E. Kavakli, S. Gritzalis, Addressing privacy requirements in system design: The PriS method, Requirements Engineering 13 (3) (2008) 241–255. doi:10.1007/s00766-008-0067-3.

[8] W. Labda, N. Mehandjiev, P. Sampaio, Modeling of privacy-aware business processes in BPMN to protect personal data, in: Proceedings of the 29th Annual ACM Symposium on Applied Computing, ACM, 2014, pp. 1399–1405.

[9] M. Gharib, P. Giorgini, J. Mylopoulos, Towards an Ontology for Privacy Requirements via a Systematic Literature Review, in: International Conference on Conceptual Modeling, Vol. 10650 LNCS, Springer, 2017, pp. 193–208. doi:10.1007/978-3-319-69904-2_16.

[10] S. D. Warren, L. D. Brandeis, The Right to Privacy, Harvard Law Review 4 (5) (1890) 193. doi:10.2307/1321160.

[11] A. F. Westin, Privacy and freedom, Washington and Lee Law Review 25 (1) (1968) 166.

[12] A. Etzioni, The Limits of Privacy, Ethics 111 (4) (1999) 288. doi:10.1086/233581.

[13] I. Altman, Privacy: a conceptual analysis, Environment and behavior 8 (1) (1976) 7–29.

[14] M. J. Culnan, P. K. Armstrong, Information privacy concerns procedural fairness and impersonal trust: an empirical investigation, Organization science 10 (1) (1999) 104–115.

[15] D. J. Solove, A Taxonomy of Privacy, University of Pennsylvania Law Review 154 (3) (2006) 477. doi:10.2307/40041279.

[16] D. Zwick, N. Dholakia, Whose identity is it anyway? Consumer representation in the age of database marketing, Journal of Macromarketing 24 (1) (2004) 31–43.

[17] A. Pfitzmann, M. Hansen,A terminology for talking about privacy by data minimization: Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management, Technical University Dresden (2010) 1–98

[18] J. Phelps, G. Nowak, E. Ferrell, Privacy concerns and consumer willingness to provide personal information, Journal of Public Policy & Marketing 19 (1) (2000) 27–41.

[19] K. B. Sheehan, M. G. Hoy, Dimensions of privacy concern among online consumers, Journal of public policy & marketing 19 (1) (2000) 62–73.

[20] H. Krasnova, S. Spiekermann, K. Koroleva, T. Hildebrand, Online social networks: why we disclose, Journal of Information Technology 25 (2) (2010) 109–125.

[21] A. Krishnan, The Personalization Privacy Paradox: An Empirical Evaluation of Information Transparency and the Willingness to Be Profiled Online for Personalization, MIS Quarterly 30 (1) (2006) 13. doi:10.2307/25148715.

[22] T. Dinev, H. Xu, J. H. Smith, P. Hart, Information privacy and correlates: An empirical attempt to bridge and distinguish privacy related concepts, European Journal of Information Systems 22 (3) (2013) 295–316. doi:10.1057/ejis.2012.23.

[23] H. Kaiya, M. Saeki, Using Domain Ontology as Domain Knowledge for Requirements Elicitation, in: 14th IEEE International Requirements Engineering Conference (RE'06), IEEE, 2006, pp. 189–198. doi:10.1109/RE.2006.72.

[24] D. V. Dzung, A. Ohnishi, Ontology-based reasoning in requirements elicitation, in: SEFM 2009 - 7th IEEE International Conference on Software Engineering and Formal Methods, IEEE, 2009, pp. 263–272. doi:10.1109/SEFM.2009.31.

[25] A. Souag, C. Salinesi, R. Mazo, I. Comyn-Wattiau, A security ontology for security requirements elicitation, in: Engineering Secure Software and Systems, Springer, 2015, pp. 157–177.

[26] S. T. Margulis, Privacy as a social issue and behavioral concept, Journal of Social Issues 59 (2) (2003) 243–261. doi:10.1111/1540-4560.00063.

[27] M. Gharib, J. Mylopoulos, P. Giorgini, COPri - A Core Ontology for Privacy Requirements Engineering, in: Lecture Notes in Business Information Processing, Vol. 385 LNBIP, Springer, 2020, pp. 472–489. doi:10.1007/978-3-030-50316-1_28.

[28] P. Rashidi, A. Mihailidis, A survey on ambient-assisted living tools for older adults, IEEE journal of biomedical and health informatics 17 (3) (2013) 579–590.

[29] M. Ziefle, C. Rocker, A. Holzinger, Medical technology in smart homes: exploring the user's perspective on privacy, intimacy and trust, in: 35th Computer Software and Applications Conference Workshops (COMPSACW), IEEE, 2011, pp. 410–415.

[30] S.Drude, Abstracting information on body area networks, Ph.D. thesis, University of Cambridge (2006).

[31] D. He, C. Chen, S. Chan, J. Bu, A. V. Vasilakos, A distributed trust evaluation model and its application scenarios for medical sensor networks, IEEE Transactions on Information Technology in Biomedicine 16 (6) (2012) 1164–1175. doi:10.1109/TITB.2012.2199996.

[32] S. Beul, M. Ziefle, E. M. Jakobs, It's all about the medium: Identifying patients' medial preferences for telemedical consultations, in: Lecture Notes in Computer Science, Vol. 7058 LNCS, Springer, 2011, pp. 321–336. doi:10.1007/978-3-642-25364-5_23.

[33] K. Yusof, K. H. B. Neoh, M. A. bin Hashim, I. Ibrahim, Others, Role of teleconsultation in moving the healthcare system forward, Asia-Pacific Journal of Public Health 14 (1) (2002) 29–34.

[34] E. A. Miller, The technical and interpersonal aspects of telemedicine: effects on doctor–patient communication, Journal of telemedicine and telecare 9 (1) (2003) 1–7.

[35] J. I. Hong, J. A. Landay, An architecture for privacy-sensitive ubiquitous computing, Proceedings of the 2nd international conference on Mobile systems, applications, and services - MobiSYS '04 (2004) 177doi:10.1145/990064.990087.

[36] M. Uschold, Building Ontologies : Towards a Unified Methodology, Proceedings Expert Systems 1996, the 16th Annual Conference of the British Computer Society Specialist Group on Expert Systems (September) (1996) 1–18. doi:10.1.1.39.9075.

[37] M. Fernández-López, A. Gómez-Pérez, N. Juristo, METHONTOLOGY: From Ontological Art Towards Ontological Engineering, AAAI-97 Spring Symposium Series SS-97-06 (1997) 33–40. doi:10.1109/AXMEDIS.2007.19.

[38] T. R. Gruber, Toward principles for the design of ontologies used for knowledge sharing, International Journal of Human-Computer Studies 43 (5-6) (1995) 907–928. arXiv:0701907v3, doi:10.1006/ijhc.1995.1081.

[39] M. Gharib, P. Giorgini, J. Mylopoulos, Ontologies for Privacy Requirements Engineering: A Systematic Literature Review, arXiv preprint arXiv:1611.10097 arXiv:1611.10097.
URL http://arxiv.org/abs/1611.10097

[40] P. Haase, H. Lewen, R. Studer, D. Tran, The neon ontology engineering toolkit, (April) (2008) 4–6. doi:10.1.1.141.4163.

[41] Y. Sure, J. Angele, S. Staab, OntoEdit: Guiding ontology development by methodology and inferencing, Proc. of the Confederated International Conferences CoopIS, DOA and ODBASE (2002) 1205–1222 doi:10.1007/3-540-36124-3_76.

[42] A. Kalyanpur, B. Parsia, E. Sirin, B. C. Grau, J. Hendler, Swoop: A Web Ontology Editing Browser, Web Semantics 4 (2) (2006) 144–153. doi:10.1016/j.websem.2005.10.001.

[43] U. Prot, M. Horridge, H. Knublauch, A. Rector, R. Stevens, C. Wroe, S. Jupp, G. Moulton, N. Drummond, S. Brandt, A Practical Guide To Building OWL Ontologies Using Prot'eg'e 4 and CO-ODE Tools, Matrix (2009) 0–27.

[44] E. Prud'Hommeaux, A. Seaborne, Others, SPARQL Query Language for RDF (Working Draft), W3C recommendation.

[45] M. S. Fox, J. F. Chionglo, F. G. Fadel, A common-sense model of the enterprise, Proceedings of the 2nd Industrial and Engineering Applications of Artificial Intelligence and Expert Systems (1993) 25–34.

[46] H. Dong, F. K. Hussain, E. Chang, Application of Protégé and SPARQL in the field of project knowledge management, Second International Conference on Systems and Networks Communications, ICSNC 2007doi:10.1109/ICSNC.2007.22.

[47] M. Aljohani, J. Blustein, K. Hawkey, Toward applying online privacy patterns based on the design problem: A systematic review, in: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Vol. 10918 LNCS, Springer Verlag, 2018, pp. 608–627. doi:10.1007/978-3-319-91797-9_43.

[48] S. Agostinelli, F. M. Maggi, A. Marrella, F. Sapio, Achieving GDPR compliance of BPMN process models, in: Lecture Notes in Business Information Processing, Vol. 350, 2019, pp. 10–22. doi:10.1007/978-3-030-21297-1_2.

[49] N. Gol Mohammadi, J. Leicht, N. Ulfat-Bunyadi, M. Heisel, Privacy Policy Specification Framework for Addressing End-Users' Privacy Requirements, in: International Conference on Trust and Privacy in Digital Business, 2019, pp. 46–62. doi:10.1007/978-3-030-27813-7_4.

[50] A. Kung, F. Kargl, S. Suppan, J. Cuellar, H. C. Pöhls, A. Kapovits, N. N. McDonnell, Y. S. Martin, A Privacy Engineering Framework for the Internet of Things, in: Data Protection and Privacy:(In) visibilities and Infrastructures, Springer, 2017, pp. 163–202. doi:10.1007/978-3-319-50796-5_7.

[51] S. Braghin, A. Coen-Porisini, P. Colombo, S. Sicari, A. Trombetta, Introducing privacy in a hospital information system, in: Proceedings of the fourth international workshop on Software engineering for secure systems - SESS '08, ACM, 2008, pp. 9–16. doi:10.1145/1370905.1370907.

[52] G. W. Van Blarkom, J. J. Borking, J. G. E. Olk, Handbook of privacy and privacy-enhancing technologies, Privacy Incorporated Software Agent (PISA) Consortium, The Hague.

[53] S. Dritsas, L. Gymnopoulos, M. Karyda, T. Balopoulos, S. Kokolakis, C. Lambrinoudakis,

S. Katsikas, A knowledge-based approach to security requirements for e-health applications, Electronic Journal for E-Commerce Tools and Applications (2006) 1–24.

[54] M. Belaazi, H. B. Rahmouni, A. Bouhoula, An ontology regulating privacy oriented access controls, in: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Vol. 9572, Springer Verlag, 2016, pp. 17–35. doi:10.1007/978-3-319-31811-0_2.

[55] A. S. Ahmadian, D. Strüber, V. Riediger, J. Jürjens, Model-based privacy analysis in industrial ecosystems, in: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Vol. 10376 LNCS, Springer Verlag, 2017, pp. 215–231. doi:10.1007/978-3-319-61482-3_13.

[56] M. Gharib, P. Giorgini, Modeling and Reasoning About Information Quality Requirements, in: Requirements Engineering: Foundation for Software Quality, Vol. 9013, Springer, Springer, 2015, pp. 49–64. doi:10.1007/978-3-319-19237-6_15.

[57] J. Bhatia, T. D. Breaux, A Data Purpose Case Study of Privacy Policies, in: Proceedings - 2017 IEEE 25th International Requirements Engineering Conference, RE 2017, 2017, pp. 394–399. doi:10.1109/RE.2017.56.

[58] A. Gerl, N. Bennani, H. Kosch, L. Brunie, LPL, towards a GDPR-compliant privacy language: Formal definition and usage, in: Lecture Notes in Computer Science, Vol. 10940 LNCS, Springer Verlag, 2018, pp. 41–80. doi:10.1007/978-3-662-57932-9_2.

[59] C. Castelfranchi, Modeling social actions for AI agents, Artificial Intelligence 103 (January 1997) (1998) 157–182. doi:10.1016/S0004-3702(98)00056-3.

[60] M. Gharib, P. Giorgini, Analyzing trust requirements in socio-technical systems: A belief-based approach, in: Lecture Notes in Business Information Processing, Vol. 235, Springer, 2015, pp. 254–270. doi:10.1007/978-3-319-25897-3_17.

[61] N. Mayer, Model-based management of information system security risk, Ph.D. thesis, University of Namur (2009).

[62] E. Paja, F. Dalpiaz, P. Giorgini, {STS}-Tool: Security Requirements Engineering for Socio-Technical Systems, in: Engineering Secure Future Internet Services and Systems, Springer, 2014, pp. 65–96.

[63] H. Mouratidis, P. Giorgini, Secure Tropos: A security-oriented extension of the Tropos methodology, Journal of Software Engineering and Knowledge Engineering 17 (2) (2007) 285–309.

[64] M. Deng, K. Wuyts, R. Scandariato, B. Preneel, W. Joosen, A privacy threat analysis framework: Supporting the elicitation and fulfillment of privacy requirements, Requirements Engineering 16 (1) (2011) 3–32. doi:10.1007/s00766-010-0115-7.

[65] S. Chen, M. A. Williams, Privacy: An ontological problem, in: PACIS 2010 - 14th Pacific Asia Conference on Information Systems, 2010, pp. 1402–1413.

[66] M. Gharib, P. Lollini, A. Bondavalli, A conceptual model for analyzing information quality in System-of-Systems, in: 12th System of Systems Engineering Conference, SoSE17, IEEE, 2017, pp. 1–6. doi:10.1109/SYSOSE.2017.7994946.

[67] M. Horridge, H. Knublauch, A. Rector, R. Stevens, C. Wroe, S. Jupp, G. Moulton, R. Stevens, N. Drummond, S. Jupp, G. Moulton, S. Brandt, A Practical Guide to Building OWL Ontologies Using Protege 4 and CO-ODE Tools, Matrix (2011) 0–107.

[68] M. Poveda-villalón, M. C. Suárez-figueroa, A Double Classification of Common Pitfalls in Ontologies, Development (2010) 1–12.
URL http://oa.upm.es/5413/

[69] G'omez-P'ereza, Asunci'on, OOPS ! (OntOlogy Pitfall Scanner !): supporting ontology evaluation on-line 1 (2009) 1–5.

[70] F. Dalpiaz, E. Paja, P. Giorgini, Security Requirements Engineering: Designing Secure Socio-Technical Systems, MIT Press, 2016.

[71] M. Gharib, P. Giorgini, J. Mylopoulos, Analysis of information quality requirements in business processes, revisited, Requirements Engineering 23 (2) (2018) 227–249. doi:10.1007/s00766-016-0264-4.

[72] P. Runeson, M. Höst, Guidelines for conducting and reporting case study research in software engineering, Empirical Software Engineering 14 (2) (2009) 131–164. doi:10.1007/s10664-008-9102-8.

[73] W. Trochim, J. P. Donnelly, The Research Methods Knowledge Base, Cengage Learning, 2006.

[74] E. S.-k. Yu, Modelling Strategic Relationships for Process, Ph.D. thesis, University of Toronto (1995).

[75] M. Palmirani, M. Martoni, A. Rossi, C. Bartolini, L. Robaldo, PrOnto: Privacy Ontology for Legal Reasoning, Electronic Government and the Information Systems Perspective (2018) 139–152 doi:10.1007/978-3-319-98349-3_11.

[76] A. Oltramari, D. Piraviperumal, F. Schaub, S. Wilson, S. Cherivirala, T. B. Norton, N. C. Russell, P. Story, J. Reidenberg, N. Sadeh, PrivOnto: A semantic framework for the analysis of privacy policies, Semantic Web 9 (2) (2018) 185–203. doi:10.3233/SW-170283.

[77] A. Singhal, D. Wijesekera, Ontologies for modeling enterprise level security metrics, in: Proceedings of the Sixth Annual Workshop on Cyber Security and Information Intelligence Research, ACM, 2010, p. 58.

[78] F. Massacci, J. Mylopoulos, F. Paci, T. T. Tun, Y. Yu, An extended ontology for security requirements, in: Advanced Information Systems Engineering Workshops, Springer, 2011, pp. 622–636.

[79] J. L. Velasco, R. Valencia-Garc\'\ia, J. T. Fernández-Breis, A. Toval, Others, Modelling reusable security requirements based on an ontology framework, Journal of Research and Practice in Information Technology 41 (2) (2009) 119.

[80] W. Kang, Y. Liang, A security ontology with MDA for software development, in: Proceedings - 2013 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, CyberC 2013, IEEE, 2013, pp. 67–74. doi:10.1109/CyberC.2013.20.

[81] A. I. Antón, J. Earp, A requirements taxonomy for reducing Web site privacy vulnerabilities, Requirements Engineering 9 (3) (2004) 169–185. doi:10.1007/s00766-003-0183-z.

[82] K. Wuyts, R. Scandariato, B. De Decker, W. Joosen, Linking privacy solutions to developer goals, in: Proceedings - International Conference on Availability, Reliability and Security, ARES 2009, IEEE, 2009, pp. 847–852. doi:10.1109/ARES.2009.51.

[83] J. I. Hong, J. D. Ng, S. Lederer, J. A. Landay, Privacy risk models for designing privacy-sensitive ubiquitous computing systems, in: Proceedings of the 2004 conference on Designing interactive systems processes, practices, methods, and techniques - DIS '04, ACM, 2004, p. 91. doi:10.1145/1013115.1013129.

[84] C. Jensen, J. Tullio, C. Potts, E. D. Mynatt, STRAP: A Structured Analysis Framework for Privacy, Tech. rep. (2005).

[85] S. Spiekermann, L. F. Cranor, Engineering privacy, Software Engineering, IEEE Transactions on 35 (1) (2009) 67–82.

[86] M. Deng, K. Wuyts, R. Scandariato, B. P. Wouter, A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements, Requirements Engineering 16 (1) (2011) 1–27.

[87] P. J. Radics, D. Gračanin, D. Kafura, PREprocess before you build: Introducing a framework for privacy requirements engineering, in: Proceedings - SocialCom/PASSAT/BigData/EconCom/BioMedCom 2013, IEEE, 2013, pp. 564–569. doi:10.1109/SocialCom.2013.85.

[88] J. Caramujo, A. Rodrigues da Silva, S. Monfared, A. Ribeiro, P. Calado, T. Breaux, RSL-IL4Privacy: a domain-specific language for the rigorous specification of privacy policies, Requirements Engineering 24 (1) (2019) 1–26. doi:10.1007/s00766-018-0305-2.

[89] M. Gharib, J. Mylopoulos, A core ontology for privacy requirements engineering (2018). arXiv:1811.12621.

Appendix A: the formalization of the CQs (SPARQL queries)

Table 5: Competency Questions and their formalization (SPARQL queries) for validating the COPri ontology

	Organizational dimension
CQ1.	Who are the delegators that delegate produce [read | modify | collect] permission, which is not accompanied by trust nor monitoring? - (returns also information and delegatees)
	`SELECT` ?actor1 ?info ?actor2 `WHERE` {?actor1 copri:delegator ?delegate. ?delegate copri:permissionDelegatum ?perm. ?perm copri:hasPermissionType copri:permProduce. [:permRead | :permModify | :permCollect] ?perm copri:over ?info. ?delegate copri:delegatee ?actor2. `FILTER NOT EXISTS` { ?actor1 copri:trustor ?trust. ?trust copri:trustee ?actor2. ?trust copri:hasTrustLevel copri:trust. ?trust copri:permissionTrustum ?perm. ?perm copri:hasPermissionType copri:permProduce. [:permRead | :permModify | :permCollect]} `FILTER NOT EXISTS` {?actor1 copri:monitor ?monitor. ?monitor copri:monitoree ?actor2. ?monitor copri:ofPermission ?perm.}}
CQ2.	Who are the delegators that delegate produce [read | modify | collect] permission, which are accompanied by both trust and monitoring? - (returns also information and delegatees)
	`SELECT` ?actor1 ?info ?actor2 `WHERE` {?actor1 copri:delegator ?delegate. ?delegate copri:permissionDelegatum ?perm. ?perm copri:hasPermissionType copri:permProduce. [:permRead | :permModify | :permCollect] ?perm copri:over ?info. ?delegate copri:delegatee ?actor2. ?actor1 copri:trustor ?trust. ?trust copri:trustee ?actor2. ?trust copri:hasTrustLevel copri:trust. ?trust copri:permissionTrustum ?perm. ?perm copri:hasPermissionType copri:permProduce. [:permRead | :permModify | :permCollect] ?actor1 copri:monitor ?monitor. ?monitor copri:monitoree ?actor2. ?monitor copri:ofPermission ?perm.}
CQ3.	Which are the delegatee that can repudiate that he/she accepted the delegatum? - (returns also delegator)
	`SELECT` ?actor2 ?actor1 `WHERE` {?actor1 copri:delegator ?delegate.

?delegate copri:hasDelegationType copri:repudiation.
?delegate copri:delegatee ?actor2.}

CQ4.	Which are the role(s) that each agent is playing?
	`SELECT` ?actor ?role `WHERE` {?actor copri:plays ?role.}
CQ5.	Which is the personal information of sensitivity Sensitive [NonSensitive]? ?
	`SELECT` ?PerInfo `WHERE` {?PerInfo copri:hasSensitivityLevel copri:slSensitive [:slNonSensitive]}
	Risk dimension
CQ6.	Which are the existing vulnerabilities and which personal information are subject to them?
	`SELECT` ?Vulnerability ?PerInfo `WHERE` {?Vulnerability copri:isSubjectTo ?PerInfo}
CQ7.	Which are the existing vulnerabilities and which are the threats that can exploit them?
	`SELECT` ?Threat ?Vulnerability `WHERE` {?Threat copri:exploits ?Vulnerability}
CQ8.	Which are the existing vulnerabilities that are not mitigated by privacy goals?
	`SELECT` ?Vulnerability `WHERE` {?Threat copri:exploits ?Vulnerability. `FILTER NOT EXISTS` {?PriGoal copri:mitigates ?Vulnerability.}
CQ9.	Which are the existing threats and which are the personal information that are threatened by them?
	`SELECT` ?Threat ?PerInfo `WHERE` {?Threat copri:threaten ?PerInfo}
CQ10.	Which are the existing threats that have an impact with severity level Low [Medium, High] over personal information?
	`SELECT` ?Threat `WHERE` {?Threat copri:hasImpact copri:SevLlowSeverity [:SevLmediumSeverity | :SevLhighSeverity]}
CQ11.	Which are the existing intentional threats and which are the personal information that are threatened by them?
	`SELECT` ?Threat `WHERE` {?Threat copri:threaten ?PerInfo. ?Threat copri:includes ?AttackMeth}
CQ12.	Who are the threat actors and which are the intentional threats that they intend for?
	`SELECT` ?ThrActor ?Threat `WHERE` {?ThrActor copri:intends ?Threat}
CQ13.	Which are the existing attack methods and to which intentional threats they can be used for?
	`SELECT` ?AttackMeth ?Threat `WHERE` {?Threat copri:includes ?AttackMeth}
CQ14.	Which are the existing incidental threats and which are the personal information that are threatened by them?
	`SELECT` ?Threat ?PerInfo `WHERE` {?Threat copri:threaten ?PerInfo. `FILTER NOT EXISTS` {?Threat copri:includes ?AttackMeth.}
CQ15.	Which are the existing incidental threats of probability Low [Medium | High]?
	`SELECT` ?Threat

WHERE {?Threat copri:hasProbability copri:pllow
[:plmedium | :plhigh]}

	Treatment dimension
CQ16.	Which are the privacy goals that are realized by privacy constraints? - (returns also privacy constraints)
	SELECT ?PriGoal ??PriCon WHERE {?PriGoal copri:realizedBy ?PriCon}
CQ17.	Which are the existing privacy mechanisms and which are the personal information that such mechanisms are applied to?
	SELECT ?PriMech ?PerInfo WHERE {?PriMech copri:appliedTo ?PerInfo}
	Privacy dimension
CQ18.	Which is the personal information that is read without read permissions? - (returns also data subject, misusing actor and the goal using such information)
	SELECT ?actor1 ?PerInfo ?goal ?actor2 WHERE {?PerInfo copri:related ?actor2. ?use copri:useOf ?PerInfo. ?goal copri:useBy ?use. ?use copri:hasTypeOfUse copri:read. ?actor copri:aims ?goal. FILTER NOT EXISTS {?actor1 copri:hasPermission ?perm. ?perm copri:over ?PerInfo ?perm copri:hasPermissionType copri:permRead}}
CQ19.	Which is the personal information that are transferred relying on non-confidential provision? - (returns also data subjects)
	SELECT ?PerInfo ?Actor WHERE {?PerInfo copri:related ?Actor. ?Prov copri:provisionOf ?PerInfo. ?Prov copri:hasProvisionType copri:nonConfidentialProv}
CQ20.	Which is the personal information that is used by a goal, where their usage (*NtU*) is not strictly required (i.e., optional)? - (returns also personal information)
	SELECT ?PerInfo ?goal WHERE {?use copri:useOf ?PerInfo. ?goal copri:useBy ?use ?use copri:hasNeedtoUseType copri:optional}
CQ21.	Which is the personal information that is used by goals, where their purpose of use (*PoU*) is incompatible with consents provided by their data subjects? - (returns also personal information, the using goal, and the misusing actor) ?
	SELECT ?actor ?PerInfo ?goal WHERE {?use copri:useOf ?PerInfo. ?use copri:hasPurposeOfUseType copri:PoUT_S [PoUT_L| PoUT_C| PoUT_P| PoUT_M| PoUT_O]. ?goal copri:useBy ?use. ?actor copri:aims ?goal. FILTER NOT EXISTS{?actor1 copri:granter ?consent.

?consent copri:grantee ?actor.
?consent copri:toUse ?PerInfo.
?consent copri:hasConsentPoU copri:PoUT_S
[PoUT_L| PoUT_C| PoUT_P| PoUT_M| PoUT_O]. }}

CQ22. Which is the personal information that can disclose the identity of their data subjects (not anonymized)? - (returns also data subjects)

```
SELECT ?PerInfo ?Actor
    WHERE {?PerInfo copri:related ?Actor.
    FILTER NOT EXISTS {copri:PC1_Anonymize copri:appliedTo ?PerInfo.}}
```

CQ23. Which is the personal information that can be linked back to their data subjects? - (returns also data subject)

```
SELECT ?PerInfo ?Actor
    WHERE {?PerInfo copri:related ?Actor.
    FILTER NOT EXISTS {copri:PC2_Unlinkability copri:appliedTo ?PerInfo.}}
```

CQ24. Which is the personal information that describes a goal, and it is also being collected by some actor? - (returns also the using goals and data subjects) ?

```
SELECT ?PerInfo ?goal2 ?Actor
    WHERE {?PerInfo copri:related ?Actor.
            ?PerInfo copri:describes ?goal1.
            ?use copri:useOf ?PerInfo.
            ?goal2 copri:useBy ?use.
            ?use copri:hasTypeOfUse copri:collect}
```

CQ25. Who are the actors that are collecting personal information without collect permissions? - (returns also personal information and data subjects)

```
SELECT ?actor ?PerInfo ?actor1
  WHERE {?PerInfo copri:related ?actor1.
          ?PerInfo copri:describes ?goal1.
          ?use copri:useOf ?PerInfo.
          ?goal2 copri:useBy ?use.
          ?use copri:hasTypeOfUse copri:collect.
          ?actor copri:aims ?goal2.
  FILTER NOT EXISTS {?actor copri:hasPermission ?perm.
          ?perm copri:hasPermissionType copri:permCollect}}
```

CQ26. Which are the *Personally Identifiable Information (PII)* that is being collected for (S)ervice [(L)egal | (C)ommunication | (P)rotection | (M)erger | (O)ther] purpose? - (returns also the using goal and data subjects)

```
SELECT ?PII ?goal ?actor1
  WHERE {?PII copri:identify ?actor.
          ?use copri:useOf ?PII.
          ?goal copri:useBy ?use.
          ?use copri:hasTypeOfUse copri:collect
          ?use copri:hasPurposeOfUseType copri:PoUT_S.
            [PoUT_L | PoUT_C | PoUT_P | PoUT_M | PoUT_O]}}
```

CQ27. Who are the actors that do not play any role or they are playing a threat actor role?

```
SELECT ?actor
```

```
WHERE {?actor rdf:type/rdfs:subClassOf* copri:Agent.
FILTER NOT EXISTS {?actor copri:plays ?role.}
UNION {?actor copri:intends ?InThreat.}}
```

CQ28. Who are the actors that are using (producing, reading, modifying, or collecting) personal information without the required permission? - (returns also information, data subjects and the using goal)

```
SELECT ?actor ?PerInfo ?actor1 ?goal2
    WHERE {?PerInfo copri:related ?actor1.
            ?use copri:useOf ?PerInfo.
            ?goal2 copri:useBy ?use.
            ?use copri:hasTypeOfUse copri:produce
            [read | modify | collect]
            ?actor copri:aims ?goal2.
    FILTER NOT EXISTS {?actor copri:hasPermission ?perm.
            ?perm copri:over ?PerInfo.
            ?perm copri:hasPermissionType copri:permProduce
            [:permRead | :permModify | :permCollect]}}
```

CQ29. Who are the delegatees that have not adopted their delegatum? - (returns also delegatum and delegator)

```
SELECT ?actor ?delegatum ?actor1
    WHERE{?actor1 copri:delegator ?delegatum.
            ?delegatum copri:delegatee ?actor.
    FILTER NOT EXISTS{?actor copri:adopts ?delegatum.}}
```