Kaniz Fatema, Ensar Hadziselimovic, Harshvardhan J. Pandit, Christophe Debruyne, Dave Lewis, and Declan O'Sullivan. Compliance through informed consent: Semantic based consent permission and data management model. In Christopher Brewster, Michelle Cheatham, Mathieu d'Aquin, Stefan Decker, and Sabrina Kirrane, editors, Proceedings of the 5th Workshop on Society, Privacy and the Semantic Web - Policy and Technology (PrivOn2017) co-located with 16th International Semantic Web Conference (ISWC 2017), Vienna, Austria, October 22, 2017, volume 1951 of CEUR Workshop Proceedings. CEUR-WS.org, 2017

Compliance through Informed Consent: Semantic Based Consent Permission and Data Management Model

Kaniz Fatema, Ensar Hadziselimovic, Harshvardhan Pandit,
Christophe Debruyne, Dave Lewis, Declan O'Sullivan

ADAPT Centre,
Trinity College Dublin, Ireland
{firstname.lastname}@adaptcentre.ie

Abstract. The General Data Protection Regulations (GDPR) imposes greater restrictions on obtaining valid user consents involving the use of personal data. A semantic model of consent can make the concepts of consent explicit, establish a common understanding and enable re-use of consent. Therefore, forming a semantic model of consent will satisfy the GDPR requirements of specificity and unambiguity and is an important step towards ensuring compliance. In this paper, we discuss obtaining an open vocabulary of expressing consent leveraging existing semantic models of provenance, processes, permission and obligations. We also present a reference architecture for the management of data processing according to consent permission. This data management model utilizes the open vocabulary of consent and incorporates the change of context into the data processing activity. By identifying and incorporating changes to the relational context between data controllers and data subjects into the data processing model, it aims to improve the integration of data management across different information systems specifically adhering to the GDPR and helping controllers to demonstrate compliance.

Keywords: informed consent; GDPR; compliance; ontology; consent and data management model; context

1 Introduction

According to the General Data Protection Regulation (GDPR), consent of a data subject is one of the legitimate basis for processing personal data [1]. However, it is argued that in practice, consent is the only lawful basis of data processing for most of the cases [2]. The GDPR has imposed greater restrictions on obtaining consent where the controller must be able to demonstrate that the consent was validly obtained. Failure to provide a proof of validity of obtained consent will be a breach of the legal requirement for GDPR compliance. This higher restriction on consent requirements has raised the need for organizations to review their consent model to be compliant with the GDPR. By complying with the GDPR requirements, businesses will benefit from avoiding legal issues while improving customer data protection and trust. In this paper, we present a semantic model for consent that helps to identify permission and provenance related important concepts of consent. The permission related concepts illustrate the legitimacy of data processing while provenance related concepts demonstrate the compliance of processing. We also present a reference architecture of Consent and Data Management Model (CDMM) that uses this consent ontology, and aims at reducing the burden on the data controller of proving the validity of data processing in the context of the relevant consent. Irish data protection commissioner published a number of cases where the data controllers were penalized due to failure of handing consent in a compliant nature [25]. While GDPR has further heighten the restriction on consent, the controllers need to be more prepared for demonstrating compliance, especially when dealing with various customers, using different combinations of online services, that operate over multiple databases, from multiple vendors or cloud providers where there are a lot of complexities involved in building and maintaining a complete model of how personal data is used. Our presented model, CDMM, addresses the lifecycles of consent and data along with the various interactions between their stages to ensure compliance of data processing activities with the consent. This model also incorporates the interactions between consent and data lifecycle states due to change of context. Identifying changes to the context of the relationship between the data controller and the data subject the model provides automated reasoning over the contextual integrity [6] of the relationship between user consent and compliance by service providers. The contributions of the paper are:

- An open vocabulary for expressing consent by leveraging existing open ontology models to improve the integration of data management across different information systems.

- Providing mapping of consent into machine readable and machine enforceable format.

- Proposing a consent and data management model by taking into account of consent and data lifecycle. Identifying and incorporating the change of context into the data processing model.

The structure of the rest of the paper is as follows: Section 2 describes the key aspects of GDPR, Section 3 describes the consent and data lifecycle, and changes of context. Section 4 describes the consent and data management model. Section 5 narrates the implementation, while Section 6 reveals the related work and Section 7 concludes the paper.

2 GDPR

Organizations dealing with personal data of EU citizens must ensure that they're compliant with the new requirements of the General Data Protection Regulation (GDPR) before it becomes effective on 2018, which is replacing the EU Data Protection Directive [3]. The controller should also be able to demonstrate the validity of the consent (Art. 5 &7). According to the GDPR Rec. 32, 42, Art. 4, 6(2), 7 and 8, to be considered as a valid one, the consent should be:

Freely given: Consent is not freely given if the data subject has no genuine and free choice, or is unable to refuse or withdraw consent easily, or there is a "clear imbalance" of power between the controller and the data subject (e.g., between an employer and an employee).

Specific: The request for consent shall be presented in a manner which is clearly distinguishable from the other matters, in an intelligible and easily accessible form, using clear and plain language.

Informed: Consent must be taken from data subject by informing the nature of processing in an intelligible format, purpose of processing and identify of the controller and information about withdrawing consent.

Unambiguous: Data subject should take clear affirmative action as an indication to acceptance to the proposed processing of personal data. Inactivity, pre-ticked boxes, silence, failure to opt-out, or passive acquiescence do not form a valid consent.

Parental permission: The processing of the personal data of a child (aged below 16) shall be lawful only if the consent is given or authorized by the holder of parental responsibility over the child. Controller is held responsible for verifying such consent.

The data subject shall have the right to withdraw his or her consent at any time. The withdrawal of consent shall not affect the lawfulness of processing based on consent before its withdrawal. Therefore, the controller should not only provide options to withdraw the consent, but also should ensure data processing accordingly and keep the provenance of data processing to demonstrate the lawfulness of processing.

3 CONSENT, DATA AND CONTEXT

Before describing the model that incorporates interactions of consent, data and context, here we are describing the lifecycle of data and consent, and the change of context.

3.1 Lifecycle of Data

Personal data goes through a number of phases [4,5]. Lifecycle of personal data starts with the phase of generation or collection. At this phase, data can be collected about individuals, e.g., during registration process for a service; or can be created by themselves, e.g., text or picture uploaded by individuals; or can be gathered from the context, e.g., time of using a service, location and device used for the service; or can be inferred other data, e.g., a person's credit score from his transaction records. There are a number of things to consider at this phase. For example, data needs to be classified accurately to provide proper protection to sensitive personal data. Collection should be limited to only what is necessary for the purpose. Before collection or generation of data consent should be obtained.

After data is collected it is transferred to the service provider's side. Once transferred it is then used. The use of information should be consistent with the purposes for which it was collected and the commitments made to the consumer by the provider. While using data, proper access control should be enforced and data needs to be managed in way that the compliance with the legal requirements can be verified.

Data can be shared with a third party. While sharing data with a third party consent of the data subject needs to be considered, access control mechanism should allow only the authorized party to access the data, while sharing data the location of data where it is going would be an important factor to consider due to compliance requirement. Before sharing personal data some transformation might be necessary, e.g., isolation of sensitive information, ensure anonymity and unlinkability.

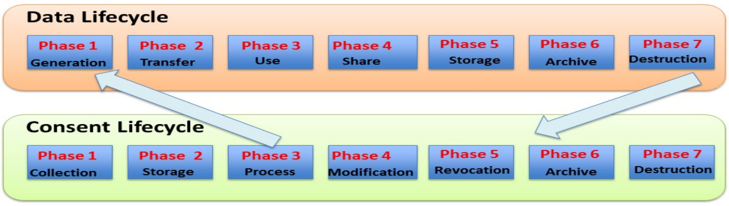

Fig. 1. Consent and data lifecycle

Data can be stored within the organization. Storage duration, purpose should adhere to the consent and should not be longer than is needed for the consented purpose. While storing, data can be encrypted. It might be difficult to process encrypted data. Other forms of transformation might be used, e.g., isolation of sensitive information, ensure anonymity and unlinkability. Proper access control should be enforced also should consider insider attack.

Data can sometimes be archived. Purpose and duration for that need to be checked against the given consent. Finally, data is destroyed. Data can be destroyed when consent is revoked, or expired.

3.2 Lifecycle of Consent

Consent goes through different phases in its lifecycle, as shown in Fig. 1. Consent is first collected, then stored and processed for checking compliance with data processing. The GDPR has clearly addressed the right of data subject to withdraw or revoke consent anytime (Art. 7, 17). Consent can also be modified, which is equivalent to revoke a previous consent and re-consent to a new one. The modification can be initiated by the data subject or due to change of context and the controller can re-solicit for consent that can lead to modification of the consent. Any change to consent, including revocation, needs to be archived for the duration necessary for verification or provenance purposes, before finally it is destroyed.

Fig. 1 shows the various phases of consent and data lifecycle and their correlation. Lifecycle of data starts only after the phase 3 of lifecycle of consent i.e. the collection of data starts only after the consent is processed to ensure the data collection compliance with the consent. Lifecycle of consent continues after the lifecycle of data finishes, to capture the provenance of consent. Any consent/ data management system should adhere to the lifecycle and their relationship.

3.3 Change of Context

Consent is given for a certain context, for e.g., for a certain service provided by a certain organization for a certain purpose. The purpose of using data might change overtime due to adoption of new technology. This change of purpose is a change of context for which consent was obtained. According to the GDPR when the intended purpose of data processing changes the controller should provide the data subject with information on that other purpose and other necessary information. This kind of context is generated from the provider's side. Another example for such context can be occurrence of data breach. Context can also be user generated, such as, modify or withdraw of data/ consent. Context can be generated from the environment, such as, expiry of consent or data or detection of other external context change. An example of external context change can be acquisition of start-ups or merging companies, which is a quite common occurrence these days [18]. In the era of cloud computing it is common to lease cloud services from various providers to form an online service. At some stage the online service provider might require to lease a service and get personal data processed from another cloud provider for which data subject did not consent at the first place. Another example is change of partner for leveraging personal data for online advertising. The change of organization or service provider for which the consent was originally obtained may not remain valid since it does not reflect the data subject's consent for the new organization that acquired his/her data along with the organization.

4 CONSENT AND DATA MANAGEMENT MODEL

In this section we describe our proposed consent and data management model. The model concentrates on two objectives:

1. To identify important concepts of consent and their interrelationships as the basis for a homogeneous representation of consent. It aims to make the consent specific, clear and unambiguous.
2. To manage data according to the consent and data lifecycle, while incorporating changes of context in the data processing model. This aims to reduce the cost and improve the efficiency of assessing and demonstrating organizations' compliance.

4.1 Consent Ontology

An ontology is commonly defined as: "an explicit specification of a conceptualization" [17]. In order to be meaningful and machine-processable, these ontologies are shared by a community of stakeholders and made explicit in some formal language such as the Web Ontology Language (OWL). OWL is an ontology language that provides support for reasoning tasks such as inferring implicit facts from explicit facts, classification of instances, and satisfiability checking (i.e., checking whether there are contradictions in the knowledge base). We use the Resource Description Framework (RDF) to store both our OWL ontology and knowledge base containing information using that ontology as RDF.

Our Consent Ontology formalizes a generic model for the notion of "consent" by identifying and describing the important concepts and relations. The basic purpose of giving consent is providing permission to perform personal data processing for specified purposes. The ontology should cover the permission-related concepts of consent such as “who is allowed or denied to do some activity on what data”. Each permission is granted for a certain time and therefore validity time is also a key part of permission of consent. A consent can also cover obligations which are the actions to be performed when certain event, such as change of context, occurs. An example of consent related obligation can be to renegotiate consent when the purpose of data processing changes. Since GDPR prescribes that it is the controller’s responsibility to demonstrate that the consent was validly obtained, the ontology should provide support for the provenance of consent obtained as well. Therefore, the ontology has concepts of consenting party and context which includes the time, location, format of gathering consent and as well as information provided when consent was requested. To incorporate these concepts in the consent ontology we extended the Provenance Ontology (PROV-O) a W3C Recommendation. PROV-O allows us to model the provenance of Entities in terms of the Activities that that generate or process them and the Agents undertaking those Activities. These provenance concepts are mapped onto the consent management ontology in Fig. 2. In this way, other extension to PROV-O (e.g., for planning processes or for specific resource types) and tools that process provenance data can be used to support the implementation of the consent management system. In future, we will align it with ODRL Permissions & Obligations Expression (POE).

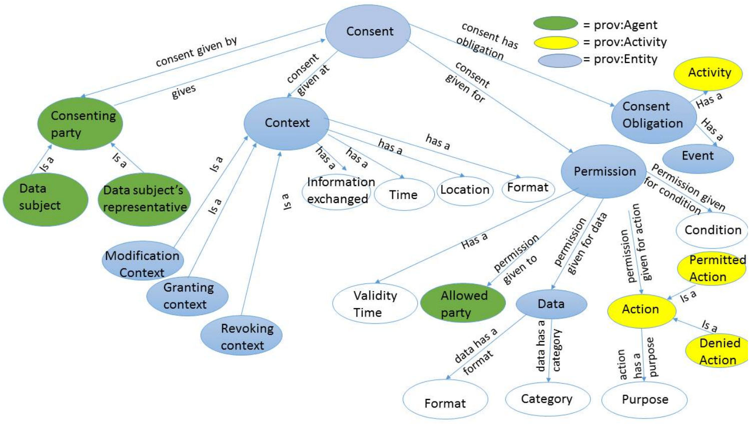

Fig. 2. Semantic model of consent

We stated that our ontology provides a generic model. With “generic”, we mean that we provide an upper common ontology for consent; one that provides a broad, reusable coverage that needs to be specialized for specific cases. As the ontology is shared – i.e., accessible via a URI that is both used as the ontology’s namespaces and resolves to the ontology document via basic Semantic Web and Linked Data principles. An upper common ontology provides the abstract concepts and relations that are shared across by many domains and applications and queries formulated in terms of those will return meaningful answers for each system. While reusable, the ontology might not be useable for very specific tasks or domains because of its under specification. The tension field between generic, reusable ontologies and specific, useable ontologies is well described in [20]. We deemed the creation of an exhaustive ontology covering all domains not feasible, and instead, propose that others should extend this ontology for specific application scenarios or application domains. With the Web Ontology Language, this is fairly straightforward by creating a new ontology, importing our consent ontology, extend the ontology and then publish it on the Web – or between the stakeholders. Another advantage of this approach is that the evolution of the extensions has no impact on the common ontology. One can adopt techniques developed in the ontology engineering community to investigate commonalities in each of those extensions for "promotion" to the common ontology, such as [19]. This approach allows different stakeholder groups to develop their own specific ontologies that would provide a better coverage (in terms of "completeness" – i.e., are all important concepts present) and granularity (i.e., the level of detail of the things that need to be represented), amongst others.

The way consent objects are presented and gathered varies from service to service and also due to the use of various devices. The ontology can be used to annotate the relevant concepts in consents that are captured in various formats. A semantically homogeneous representation can be created from these annotations and this is useful in number of ways. Firstly, it helps in getting enforceable rules and provenance information. Secondly, it helps to produce an intelligible and user friendly representation of consent by highlighting the important concepts of consent. Finally, it will further help in the migration of consent from one domain to another while ensuring the enforcement of consent.

4.2 Consent and Data Management Model (CDMM)

In this section, we present the Consent and Data Management Model (CDMM), an abstract reference architecture, to incorporate the consent permission and change of context into data processing model. It aims to reduce the burden on organisation to assess and demonstrate their compliance. The CDMM, as shown in Fig.3, consists of five components: User Interaction Handler (UIH), Consent Manager (CM), Context Handler (CH), Data Manager (DM) and Provenance Manager (PM).

The UIH is an interface for interacting with the users and is responsible for getting consent, exchanging of required information. It further provides facilities for enforcement of rights, such as, data access, erasure or rectification of data, or requests for portable data. These rights are mandated by the GDPR, are common for all data subjects, and do not depend on individual consent. However, the type of service and the scope of consent influence the implementation and precise scope of the rights. The interface used to handle such rights also should not be significantly different from the data collection interfaces. Therefore, the UIH is delegated the responsibility to handle all user actions through a single, uniform point of access and provides a uniform interface for all the users to enforce those rights.

The permissions that each data subject provides in the form of consent is distinct for each user and needs to be obtained before getting and processing the data. The CM utilizes a consent ontology to translate the obtained consent into usable properties such as *consent validity*, *consent obligation* and *consent permission*. The separation of such consent-based information is designed to ease the processing of consent, due to the differences in how these might be implemented, e.g. validity might just be some constrain check, perhaps done as deontic logic reasoning and queries directly on the owl instances, permission would map onto the XACML system, while obligation might need something different, more like a workflow management system. *Consent obligations* contain instructions related to consent, including the range of contexts under which a new consent must be sought from the data subject or under which condition the subject needs to be informed. Some other consent obligations are to stop data processing when consent expires, to delete data after a certain period, or when the purpose for which consent was sought will no longer be undertaken, and to inform data subjects whenever a data breach is detected. *Consent permission* states the processing or activities that are allowed to take place by data subjects on the personal data.

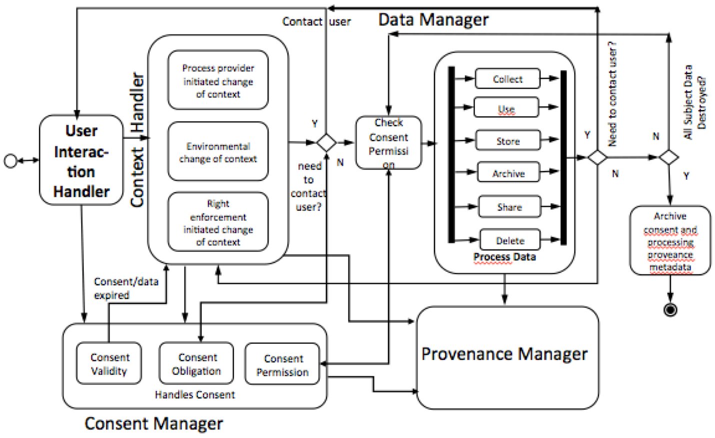

Fig. 3. Consent and data management model

The CM only keeps the up-to-dated consent, which is relevant for the current permissions for processing of data. Additional information, such as, who signed the consent, when was the consent given, what information was exchanged while providing consent is stored by the Provenance Manager (PM). The Context Handler (CH) is responsible for managing context and for detecting changes of context and informing the CM and DM of these changes. The DM is responsible for managing data according to the consent and provides protective control by ensuring only authorized processing can occur.

The PM is responsible for maintaining a provenance log of all activities involving data and consent. It tracks the lifecycles of consent and data. From the tracks it is possible to identify the source, usage and storage in data activities. Provenance logs are beneficial in compliance queries as they enable tracking of whether usage of data was according to the consent given at the time. They also can demonstrate retroactively how activities handled a change in consent and therefore were compliant with the permissions set forth by the user consent. Additionally, provenance logs maintain information about actions carried out and the resulting changes made by obligatory rights such as informing users of a data breach or when users request data rectification.

The PM aims to provide detective control for ensuring compliance. The CM enforces the correct behaviour of data handling systems according the consent and terms of GDPR, while the PM supports demonstrating that the correct behaviour has been undertaken for compliance reporting purposes.

The CDMM, however, does not concern with the data representation format. The authorization of the DM works with any data model as long as the data can be identified with an identifier [16]. It has been tested before with data in file format where the file name works as an identifier and data in RDBMS format where the primary key of table works as an identifier for data.

4.2.1 Interactions between CDMM components

The flows of interactions of the components are given below:

1. Consent is obtained via UIH before first-use of the service, such as when someone registers or signs in for the first time. The Consent Manager (CM) then annotates the consent using the consent ontology. The CM extracts meaningful information from the annotated consent and stores the *consent validity*, *consent obligations* and *consent permissions*. It also sends the provenance related information to the PM.
2. The Context Handler (CH) generates a context of arrival of a new consent (i.e. a new sign up and consent). It informs the DM and the DM can become aware of that. In case of automated data collection the DM can initiate the process and checks permission as mentioned in step 3.
3. The DM ensures adequate permissions for intended usage and processing of data exist, as specified by the consent, before any processing of data, including collection, starts.
4. When the consent is changed or updated, CH gets the context of the consent modification and informs the CM and DM. The CM identifies the change and updates the processing related information, which was created in step 1 and sends the new consent to the PM. The DM then halts the current processing and checks the new updated permissions for further processing of data.
5. A context change occurs when the intended data entity or activity changes. For example, when storage of data might shift to a different cloud storage provider, or a purpose of data collection or usage changes. An event is created based on the change and the CH is informed manually of this change. The CH then checks the consent obligation from the CM to see if it needs to interact with the user to obtain an updated consent based on the context change. If it identifies the need for a new context, it initiates interaction with the user according to the obligation provided earlier with the consent.
6. When consent is revoked, the CH informs the CM, which updates the processing information it contains to restrict access to affected data. The CH also informs the DM, which then checks the updated permission and adjusts the data collection processes accordingly. This may result in cessation of data collection based on the specific permissions and processing done by the activity. Depending on the policy of the organization it then deletes data or archives it for validity or provenance purpose.

7. The PM keeps a log of all activities involving consent and data. It records how consent was obtained based on the mechanism used by the UIH to interact with the user, the consent itself from the CM, and the activities using the consent. It also keeps a track of the data lifecycle as provisioned by the DM including activities such as storage and sharing. A record of archived consent and data is also maintained by the PM in events as described in Steps 5 & 6.

4.2.2 Use Case Scenario

Let us consider a practical use case scenario of obtaining consent for a research project of Trinity College. A consent form is used for obtaining the consent from the Consultant Psychiatrists for interviewing them about the e-referral system for mental health service. The consent form contains information of the researcher who is collecting the data, the information about the research, procedures of the study, where and how the data will be used and a declaration which clearly states what the interviewee is agreeing to, what rights s/he possesses and the participant's name, signature and date. The form contains the contact details of the researcher at the end. The declaration which clearly states what the interviewee is agreeing to may contain a number of options for the interviewee to choose from. In our example use case scenario the participant may choose to audio record the interview or not to audio record the interview. They are further given option to choose about using direct quote from their interview data, the options for that include i) to allow to use direct quote from their dialogue anonymously, with further review from participant ii) to allow to direct quote from their dialogue anonymously without further review from participant iii) not allowing to use direct quote in this project.

The permissions mentioned above are encoded in machine readable format and stored as consent permission by the CM. When the researcher wants to collect interview data s/he asks the DM if s/he can audio record the data. The DM consults the permissions the participant provided via the interface to the CM and returns a decision. If the decision is 'yes' then s/he audio records the data. While processing the data the researcher asks the DM for the data for which there is permission for using direct quote from the dialogue without reconfirmation. The DM consults the policies of the participants and returns the data for which there are such permissions. The requests and responses are logged by the PM and can be checked to verify for what condition the data was provided to the researcher. While processing the data which allows to use direct quote from the dialogue with reconfirmation, the DM instructs the researcher to contact the participants via the UIH (note. this obligation to contact participant is an obligation related to access control [16, 21]). When a user enforces a right such as 'remove from study' a context is generated upon receiving of that request by the CH. The CH informs the DM for removing the data and the CM about the modification of consent. The data is deleted by the DM and the consent is updated accordingly by the CM. The PM keeps information of when the data removal request was obtained, when and what changes are performed by the DM and CM regarding data processing and consent. If any publication of data was made before removal request was received it would be possible to prove the compliance of that processing based on consent valid for that time. At the end of the research project all data are destroyed and the consent and processing provenance data are archived which can be queried to prove compliance. We shall in future look into complex scenarios such as sharing or re-using data in a distributed cloud environment.

4.3 Consent Annotation and Representation

To assure GDPR compliance one requires to check consent permission before processing data. For the automated checking of consent permission it is needed to encode or translate the permission related information from user's consent into machine enforceable rules. We previously used templates for consent and converting them into machine enforceable rules [16]. However, this approach is highly dependent on application and use case scenario. We are now focusing into more generic solution to that by using the ontology of consent to annotate the concepts of consent with scenario specific properties. From this annotated consent we can have a mapping to machine enforceable rules in a policy language according to the choice of the controller.

In order to choose the format for encoding consent we are considering two different languages for the representation of consent permissions, which are XACML and ODRL.

XACML is OASIS standards for attribute-based access control policy language. It consists of three main elements in the hierarchical structure: PolicySet, Policy and Rule. Further, rules have Targets, Conditions and Obligations. However, the real power of the language comes from its rich set of Functions. XACML is XML-centric language. ODRL, on other hand, does not have strict rules when it comes to the output. ODRL is Rights Expression Language (REL) developed to express rights, rules and conditions, including permissions, prohibitions, obligations and assertions.

When it comes to our specific use-case in annotating the consent, both languages have their strengths and weaknesses. XACML can clearly enforce the policy rule based on the consent. While it can obviously define that the certain statement has certain action associated with it, it fails to define abstract actions, such as 'understanding', that are not particularly associated with actions, but require user to absorb the information, with no particular consequence in failing to do so.

ODRL has Classes, Concepts and Properties to describe almost any aspect of consent. It even has specific Property consentingParty and related Concept (action) obtainConsent. But it lacks straight-forward approach of XACML and might over-complicate certain aspects of the mapping process. When it comes to enforcement of permission, XACML has a rich mechanism for that and there are enforcement engine available for that. In that respect, and for the purpose of this paper, we use XACML as a base for our annotation of consent permission, due to its enforcing nature and clear rule definition. ODRL would be used for scenarios where XACML falls short, such as already mentioned generic concepts that presented in informal form.

5 IMPLEMENTATION

We have implemented the preliminary version of the ontology in Protégé which can be found in https://openscience.adaptcentre.ie/ontologies/consent. Note that our ontology imports and extends PROV-O. For more information how our concepts relate to PROV-O, we refer back to Fig. 2. This version of the ontology, as discussed in Section 4, is meant to be generic and is subject to extension for specific application scenarios or application domains.

We are working on a number of real life examples of consent to validate our model. For example, consent for medical examination, consent for participating in research experiments and consent for the usage of an online service. The CM of our model is responsible for annotating the consent collected via UH. In this phase of implementation, we have manually annotated the consent examples with the elements from consent ontology to get a machine-readable version of consent, which can be invoked by the other components of CDMM automatically.

Fig. 4 shows an example of how one can annotate of a sentence in RDFa[1], which is a W3C Recommendation for embedding RDF in XHTML documents, from a consent form that is used by a research project in Trinity College for obtaining the consent from the Consultant Psychiatrists for interviewing them about the e-referral system for mental health service. The sentence from consent that is used in this example is "I agree that my data will be used for scientific purposes". The embedded annotations capture how one gives permission to use (Permitted Action) my data (Data) for scientific purposes (Purpose). We used an RDFa parser [2]to get RDF from the annotated XHTML document which is presented in Fig. 5. Then we used XSPARQL [24] to transform the RDF into XACML XML, the codes can be found in purl.org/adaptcentre/openscience/projects/CDMM/consent. XSPARQL allows us to query the RDF and instantiate XML templates based on the results of these queries. The resulted XACML is presented in Fig. 6

```
<p typeof="consent:Permission">
 I agree that my <span rel="consent:givenFor"><span typeof="consent:Data">
 <span property="rdfs:label">data</span><span rel="consent:hasA" typeof="consent:Category">
 <span property="rdfs:label" content="interview data"></span></span></span></span> will be
 <span rel="consent:givenFor"><span typeof="consent:PermittedAction">
 <span property="rdfs:label" content="use">used</span>
 <span rel="consent:hasPurpose">for
 <span typeof="consent:Purpose"><span property="rdfs:label">
 scientific purposes</span></span></span></span></span></p>
```

Fig. 4. Annotation of consent with elements of ontology in RDFa.

```
<rdf:RDF xmlns:consent="http://theme-e.adaptcentre.ie/consent/ont.rdf#"
..
 <consent:Permission><consent:givenFor><consent:Data> <consent:hasA> <consent:Category>
          <rdfs:label>interview data</rdfs:label>
 </consent:Category></consent:hasA><rdfs:label>data</rdfs:label</consent:Data></consent:givenFor>
   <consent:givenFor><consent:PermittedAction>
       <rdfs:label>use</rdfs:label>
       <consent:hasPurpose><consent:Purpose> <rdfs:label>scientific purposes</rdfs:label>
</consent:Purpose></consent:hasPurpose </consent:PermittedAction> </consent:givenFor>
```

[1] RDF in Attributes (RDFa): https://www.w3.org/TR/xhtml-rdfa-primer/

[2] https://github.com/RDFLib/pyrdfa3

```
  </consent:Permission> </rdf:RDF>
```

Fig. 5. RDF representation of the annotated XHTML.

```
      <Rule RuleId="ConsentPolicy" Effect="Permit">
        <Condition>   <Apply FunctionId="urn:..function:and">
            <Apply FunctionId="urn:..:string-at-least-one-member-of">
              <ActionAttributeDesignator AttributeId="Purpose"
                                 DataType="http://..string"/>
              <Apply FunctionId="urn:..:string-bag">
                <AttributeValue DataType="http://..string">scientific purposes</AttributeValue>
              </Apply>   </Apply>
            <Apply FunctionId="urn:..:string-at-least-one-member-of">
              <ResourceAttributeDesignator AttributeId="urn:..:resource-id"
                                  DataType="http://..string"/>
              <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-bag">
                <AttributeValue DataType="http://..string">interview data</AttributeValue>
              </Apply>  </Apply>
            <Apply FunctionId="urn:..function:string-at-least-one-member-of">
              <ActionAttributeDesignator AttributeId="urn:..:action:action-id"
                                 DataType="http://..string"/>
              <Apply FunctionId="urn:..:function:string-bag">
                <AttributeValue DataType="http://..string">use</AttributeValue>
              </Apply>   </Apply>  </Apply> </Condition>  </Rule>
```

Fig. 6. Resulting XACML rules of applying an XSPARQL template on the distilled RDF.

The ODRL representation of the annotated example of consent of fig. 4 is presented in fig 7 (a). In this instance, the participant assigns the action 'use' to the researcher for the purpose of 'scientific purpose'.

```
<http://consent_form>
        a odrl:Rule ;
        odrl:permission [
            a odrl:Permission ;
            odrl:action odrl:use ;
            odrl: purpose  "scientific purposes";
            odrl:assignee <http:/researcher> ;
            odrl:assigner <http://participant>  ;
            odrl:target <http://interview_data> ] ;

(a)
```

```
<http://consent_form>
        a odrl:Rule ;
        odrl:action [
            a odrl:Action ;
            odrl:read odrl:reviewPolicy ;
            odrl:assigner <http://researcher> ;
            odrl:assignee <http://participant>  ;
            odrl:target <http://resource>
            dc:description "I understand that I make
illicit activities known"  ] ;
(b)
```

Fig. 7. ODRL representation of annotated consent elements.

ODRL offers to express elements of consent for cases XACML can't express. An example of consent, "I understand that I make illicit activities known", which is taken from our consent example, does not form XACML, but makes ODRL.
In the above example, fig. 7 (b) the assigner (researcher) uses the 'read' and 'reviewPolicy' ODRL concepts to require the user to read and 'understand' the particular policy.

This XAMCL rule – we note we excluded the whole XACML file for brevity – is used in an authorization system which we implemented earlier [16, 20-23], for checking the authorization permission before allowing to process the data. The DM of

CDMM (mentioned in Section 4.1) will make use of the authorization system. In our authorization model the permission of consent will form the policy for data subjects' Policy Decision Point (PDP). The logs from the authorization system will be used by the Provenance Manager (PM). Compliance can be checked as required or periodically using provenance logs maintained by the PM at required or periodic times. The detail of the PM is out of scope for this paper. We are currently working on the full prototype implementation of our reference framework which requires implementing the Context Handler (CH) and interactions of the CH with the DM and CM, design of querying the provenance information and converge that into compliance related information.

Our consent management model will further contain the enforceable policy from the GDPR, instead of the EU DPD which we had previously encoded in our system, by extending our previous experiment [15] for the GDPR. This will require to have the GDPR encoded into CNL manually and having enforceable rules from the CNL automatically. For speeding up the process of conversion we would need to check which of the rules in the EU DPD still exist in the GDPR for re-using those rules. Having enforceable rules from the GDPR will help for automated assurance of the GDPR compliance for the rules, which could eventually reduce the burden of demonstrating compliance through automation.

6 RELATED WORK

Grando and Schwab [7] have presented a "permission ontology" to capture consent permission for research studies. Our permission ontology element is inspired by their work. However, our focus is not only ensuring permission related compliance but also providing verifiability of consent through provenance. Wuyts et al. [8] have proposed an architecture to integrate patient consent in e-health access control. They capture consent as Policy Information Point (PIP), where consent is stored in the database. O'Keefe et al. [9] have presented an eConsent model and demonstrator used to investigate the implementation of patient consent as a means of controlling access to electronic health information shared among healthcare providers. Heinze, et al. [10] have proposed a consent management service to receive and store consent documents. It consists of a consent creator service to generate XACML based policies which are queried for getting consent based answers. Mont et al. [11, 12] have provided a reference model for the management of consent and revocation in an enterprise environment. None of the above work considered the change of context or did not consider the need for proving validity of consent or provenience of consent.

Asghar, and Russello [13, 14] have presented a goal driven approach to glue together authorization policies having a specific context. They used template for context and that template is filled for actual information when known. Our work differs from their approach. We focus on validity of consent when a change of context is detected such as the organization which collected the consent is taken over by another organization. When such a change of context is detected the validity of consent is checked and a decision to whether to re-solicit consent.

7 CONCLUSION AND FUTURE WORK

In this paper, we proposed a consent ontology leveraging existing models of provenance, processes, permission and obligations. The proposed consent ontology forms a semantic model of consent to make it specific and unambiguous as required by the GDPR. We have implemented the preliminary version of the ontology in Protégé. We further have proposed a reference architecture, called CDMM, for the management of consent and data which embodies a bunch of design decisions beyond the ontology and incorporates the change of context within the data processing model. The reference architecture aims to aid the controller to assess and demonstrate their compliance efficiently. We have illustrated a proof of concept by annotating consent examples with the consent ontology and demonstrated a way to obtain machine readable and enforceable format of consent using ORDL and XACML. We used the XACML obtained from consent into a policy engine which is implemented by augmenting Sun's implementation of XACML [24]. We are in the process of further refining the design of the components of CDMM towards developing a complete implementation of the prototype which will be evaluated.

Acknowledgment. This paper supported by the ADAPT Centre for Digital Content Technology, which is funded under the SFI Research Centres Programme (Grant 13/RC/2106) and is co-funded under the European Regional Development Fund.

8 REFERENCES

[1] General Data Protection Regulations (GDPR), http://ec.europa.eu/justice/data-protection/reform/files/regulation_oj_en.pdf

[2] ‘A guide for in-house lawyers’, Hunton and Williams, April 2016, https://www.huntonprivacyblog.com/wp-content/uploads/sites/18/2015/06/Hunton-Guide-to-the-EU-General-Data-Protection-Regulation.pdf

[3] European Union Data Protection Directive, Directive 95/46/EC .

[4] Chen, D., Zhao, H.: Data Security and Privacy Protection Issues in Cloud Computing. In: 2012 International Conference on Computer Science and Electronics Engineering (ICCSEE), vol.1, pp. 647-651. Hangzhou, China (23-25 March 2012).

[5] Mather, T., Kumaraswamy, S. and Latif, S.: Cloud security and privacy: an enterprise perspective on risks and compliance. In. O'Reilly Media, Inc. (2009).

[6] Nissenbaum, H.: A Contextual Approach to Privacy Online. Daedalus 140 (4), 32-48 (2011).

[7] Grando, A. and Schwab, R.: Building and evaluating an ontology-based tool for reasoning about consent permission. In. AMIA annual symposium proceedings. Vol. 2013. American Medical Informatics Association, (2013).

[8] Wuyts, K., Scandariato, R., Verhenneman, G., Joosen, W.: Integrating patient consent in e-health access control. In. Developing and Evaluating Security-Aware Software Systems, IGI Global, pp. 285-308. (2013).

[9] O'Keefe, C. M., Greenfield, P., and Goodchild, A.: A decentralised approach to electronic

consent and health information access control. Journal of Research and Practice in Information Technology 37 (2), 161-178 (2005).

[10] Heinze, O., et al.: Architecture of a consent management suite and integration into IHE-based regional health information networks. BMC medical informatics and decision making,11- 58. (2011).

[11] Mont, M. C., et al.: A conceptual model for privacy policies with consent and revocation requirements. In. IFIP PrimeLife International Summer School on Privacy and Identity Management for Life, Springer Berlin Heidelberg, (2010).

[12] Mont, M. C., et al.: On the management of consent and revocation in enterprises: setting the context. HP Laboratories, Technical Report HPL-2009-49,(2009).

[13] Asghar, M. R., and Russello, G.: Actors: A goal-driven approach for capturing and managing consent in e-health systems. In. Policies for Distributed Systems and Networks (POLICY), IEEE International Symposium on. IEEE, (2012).

[14] Russello, G., Dong, C. and Dulay, N.: Consent-based workflows for healthcare management. Policies for Distributed Systems and Networks, 2008. POLICY 2008. IEEE Workshop on. IEEE, (2008).

[15] Fatema, K., Debruyne, C., Lewis, D., OSullivan, D., Morrison, J. P. and Mazed, A. A.: A Semi-Automated Methodology for Extracting Access Control Rules from the European Data Protection Directive. In. 2016 IEEE Security and Privacy Workshops (SPW), pp. 25-32.San Jose, CA (2016).

[16] Fatema K.: Adding Privacy Protection to Policy Based Authorisation Systems. , PhD thesis, 2013, https://kar.kent.ac.uk/47905/

[17] Gruber, T.:Toward principles for the design of ontologies used for knowledge sharing. International Journal of Human-Computer Studies, 907– 928, (1993).

[18] https://techcrunch.com/tag/mergers-and-acquisitions/

[19] de Moor, A., Leenheer, P. D., and Meersman, R.: DOGMA-MESS: A Meaning Evolution Support System for Interorganizational Ontology Engineering. In. Conceptual Structures: Inspiration and Application, 14th International Conference on Conceptual Structures, ICCS, pp.189-202, Aalborg, Denmark, July 16-21, (2006).

[20] Spyns, P., Meersman, R. and Jarrar, M.: Data Modelling versus Ontology Engineering. SIGMOD Record 31(4), 12-17 (2002).

[21] Chadwick, D. W., and Fatema, K.: An advanced policy based authorisation infrastructure. In. Proceedings of the 5th ACM work-shop on Digital identity management, DIM'09, pp.81-84, Chicago, Illinois, USA, (2009).

[22] Fatema, K., Chadwick, D.W. and Lievens, S.: A Multi Privacy Policy Enforcement System. In. Privacy and Identity, IFIP AICT 352, pp. 297–310.(2011).

[23] Fatema, K. and Chadwick, D.: Resolving Policy Conflicts - Integrating Policies from Multiple Authors. In. CAiSE International Workshops, Thessaloniki, Greece, (2014).

[24] Bischof, S., Decker, S., Krennwallner, T., Lopes, N., Polleres, A.: Mapping between RDF and XML with XSPARQL. J. Data Semantics 1(3), 147-185 (2012)

[25] https://www.dataprotection.ie/docs/CASE-STUDIES-2013/1441.htm