Legal and Technical Perspectives in Data Sharing Agreements Definition

Claudio Caimi[1], Carmela Gambardella[1], Mirko Manea[1], Marinella Petrocchi[2(✉)], and Debora Stella[3]

[1] Hewlett-Packard Italiana, Milan, Italy
{claudio.caimi,carmela.gambardella,mirko.manea}@hpe.com
[2] IIT-CNR, Pisa, Italy
marinella.petrocchi@iit.cnr.it
[3] Bird and Bird, Milan, Italy
Debora.Stella@twobirds.com

Abstract. An electronic Data Sharing Agreement (DSA) is a human-readable, yet machine-processable contract, regulating how organizations and/or individuals share data. In this paper, we shed light on DSA engineering, i.e., the process of studying how data sharing is ruled in traditional legal human-readable contracts and mapping their fields (and rules) into formats that are machine-processable, leading to the transposition of the traditional contract into the electronic DSA. Tangible creation of the electronic DSA is possible through the design and implementation of a three-step DSA definition phase, with an associated authoring tool. The tool is specifically tailored for encoding not only the terms of law but also the rules that an organization may have put in place (e.g., corporate internal policies, or privacy policies, or data processing agreements) to manage the data, as well as end users' privacy preferences.

1 Introduction

Sharing data among groups of organizations and/or individuals is essential in a modern web-based society, being at the very core of scientific and business transactions. Data sharing, however, poses several problems including trust, privacy, data misuse and/or abuse, and uncontrolled propagation of data. In this paper, we focus on preserving privacy whilst sharing data based on electronic Data Sharing Agreements (DSA).

An electronic DSA is a human-readable, yet machine-processable contract, regulating how organizations and/or individuals share data. Figure 1 sketched the basic components of a DSA lifecycle.

The *template definition* stage is a preliminary phase in which a pool of available DSA templates is created, according to the purpose of the data sharing and

The research leading to these results has received funding from the EU Seventh Framework Programme (FP7/2007–2013) under grant no 610853 (Coco Cloud).

B. Berendt et al. (Eds.): APF 2015, LNCS 9484, pp. 178–192, 2016.
DOI: 10.1007/978-3-319-31456-3_10

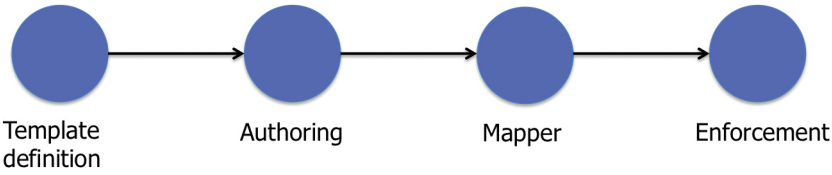

Fig. 1. Basic DSA lifecycle

the classification of data whose sharing is regulated by the DSA. The *authoring* stage is an editing tool-assisted phase, during which the stakeholders prepare the DSA itself. The result of the authoring stage is an electronic, still human readable, DSA document. The data sharing rules in the DSA are then translated to a set of enforceable security policies during the *mapper* stage. The *enforcement* stage is the phase in which the DSA is enacted on the specific data being shared.

DSA are the digital transposition of traditional legal contracts to regulate the terms and conditions under which organisations and individuals agree to share data. A first key problem in the digital world is that the constraints expressed in such (not digital) contracts remain inaccessible from the software infrastructure supporting the data sharing and management processes and, consequently, they cannot be automatically enforced. This is mainly because fields in the contract, such as its validity, the involved parties, the kind of data to be shared, and the data sharing rules themselves, still need to be interpreted and translated (primarily by humans) into meaningful technical policies, to ensure degrees of enforcement and auditing.

Overall, DSA definition, i.e., the process of creating the machine-processable document by starting from the traditional paper contract regulating data sharing, is a complex task. The main obstacles that currently prevent its successful achievement are that the process is prone to error and quite reductive (e.g., from the end users' point of view).

On the one hand, transposition of paper contracts into digital DSA involves a deep understanding of jurisdictional and intra/inter-organisations matters: as an example, legal constraints may vary from country to country, and the enactment of data sharing rules defined, e.g., at organisational level, could be subject to contextual conditions that must be considered and evaluated case by case. Traditional legal contracts may summarise in few words a series of exceptions that, if fulfilled, may change the effect of the data sharing (either allowing or denying the data access, for example). If such shades are not adequately expressed and represented in the DSA, the software infrastructure responsible for the DSA enforcement will uselessly operate.

On the other hand, standard online forms highly reduce the capabilities of end users to control how their data can be shared, by whom, and for which purposes. Usually, when end users' data is going to be processed by organisations, end users are asked to accept online the terms that will govern the data processing, by simply clicking on buttons like "Review and Accept the Terms

and Conditions" or "I accept the privacy policy". Furthermore, end users find it difficult to understand these terms and conditions, how their data will be shared practically, and how to express their potential preferences in terms of data sharing and handling. This introduces burdens on users and usability issues of solutions for end-to-end automation of contract definition and enforcement.

In this paper, we study traditional data sharing contracts with the aim of creating the corresponding enforceable data sharing agreements. In particular, we contribute by (i) evaluating which fields in a traditional data sharing contract are representable in an electronic data sharing agreement, and under which format; and (ii) showing the design and implementation of an authoring tool satisfying the representation of such fields in such formats. The authoring tool supports the creation of the contract, by leveraging on a vocabulary of reference, which defines the terms and the reference context (ontology). The tool supports a process defined on three levels of interaction and it assists an even a non-technical author (e.g., a legal expert) in the definition of the data sharing rules. Finally, the tool provides the possibility to get the user preferences and tune some rules in a final step of editing.

Roadmap. This paper is structured as follows. Section 2 expands the concept of DSA and gives a panoramic view on tools for editing privacy policies. Section 3 gives an example of legal terms contained in a traditional contract and defines fields and formats of the corresponding electronic DSA. In Sect. 4, we present architecture and functionalities of the authoring tool, through which formation of a DSA is enacted within a three-step editing phase. Finally, we conclude the paper in Sect. 5.

2 Background and Related Work

As highlighted in [5], "sharing data among groups of organizations and/or individuals is a key necessity in modern web-based society and at the very core of business transactions". A DSA is an agreement between two or more parties who wish to exchange data in several specific domains and contexts: it regulates which data to use, for which purposes and how to use it. Basically the aim of DSA is (i) to capture the data sharing policies that restrict both the party(-ies) that provide(s) the data and the party(-ies) that receive(s) the data, and (ii) to govern the data flow between them, see, e.g., [16]. Furthermore, a DSA also defines the legal obligations and the organisation policies according to, e.g., the data classification. The data classification allows to distinguish between personal data - both common data, e.g., contact details, and data belonging to special categories, like medical data - and non personal data, as, e.g., administrative and business data. In recent years, DSA have become common both in the scientific community, see, e.g., [17], and between enterprises that want to share data. In the scientific world, the importance of sharing information and making it available within the community also clashes with the assurance need that data privacy and confidentiality must be maintained. Likewise, organizations (both in the private and in the public sector) share data, on the base of agreements: nowadays, any

recipient of services must sign a contract with the provider. The goal is not only to protect the latter from improper or unauthorized usage of those services, but also to protect the recipient with respect to the misuse the provider could do over the data the recipient has provided when signing the agreement (or s/he provides and produces, using the services over the duration of the agreement). Thus, beyond defining relationships among the involved parties, an agreement also is a tool for managing risk and liability.

Initially, DSA was the electronic implementation of the common contracts stipulated in any relationship of sharing data. Currently, we assist to an evolution of electronic DSA towards not only the description of policies that govern the data sharing, the parties involved in the contract, the period of validity and other legal and business information, but also towards the DSA automatic enforcement and the verification of the effective compliance of the parties to the agreement. Thus, the rich framework for the management and the enforcement of the DSA includes authoring tools, to guide the users in the creation of comprehensive and consistent DSA; repositories to facilitate the authoring of DSA, even starting from a catalogue of DSA templates, and to manage the life-cycle management of DSA; agents to monitor and enforce the terms of the DSAs, when the checks are programmable and automatically verifiable, see, e.g. [16,17]. Contributing to move to that direction is main goal of this paper, that concentrates on the evolution of electronic DSA supported by a devoted authoring tool.

Policy Authoring Tools. Series of work in [3,4,8,9,15] connect policy authoring tools with the capability of common users to use them. In [9], the authors carry out a laboratory evaluation of a variety of user-centered methods for privacy policies authoring, to identify which design decisions should be taken for flexible and usable privacy enabling techniques. Work in [3] continues this line of research, by providing a parser which identifies the privacy policy elements in rules entered in natural languages: identification of such elements is a key step for subsequent translation of natural sentences in enforceable constructs (such as the XACML language [14]). Authors of [15] recall security and privacy policy-authoring tasks in general, and discover further usability challenges that policy authoring presents. In [4] the authors present the Coalition Policy Management Portal for policy authoring, verification, and deployment, with the goal of providing "easy to use mechanisms for refining high-level user-specified goals into low-level controls". Recently, work in [8] advances the notion of template-based authoring tools, for users with different roles and different skill sets, such as, e.g., patients, doctors, and IT administrators could be in a e-health scenario. The authors propose different templates to edit privacy policies, each of them needing different user skills to compose high-quality policies.

The FP7-EU project *Consequence* (Context Aware Data Centric Information Sharing) designed and developed an integrated framework for the authoring, analysis, and enforcement of DSA. The authoring tool developed within the project was intended for users with some knowledge on policy specification, see, e.g., [6,13]. The insertion of a help-on-line facility partly mitigates usability issues, whose complete solution needs however further investigation.

From a business perspective, Axiomatics [2] offers an authorization framework based on the XACML standard [14], that covers all the phases of the policy life-cycle, including policy creation, exploiting a graphical user interface for policy authoring, validation, deployment and enforcement.

From a social networking perspective, we may cite work in [18], which presents a *collaborative* authoring tool, allowing several individuals to specify policies over data published on social networks, and whose disclosure may affect their privacy. The authors acknowledge some usability issues in their prototype implementation, and future work are foreseen towards a user-friendly authoring interface.

3 DSA: From Legal Contracts to Machine-Processable Agreements

An electronic Data Sharing Agreement (DSA) regulates how organizations and/or individuals share data. It can be between two organisations and/or individuals (bilateral agreement), or more (multilateral agreement). DSA can also be adopted to share information inside an organisation, e.g., between its different business units.

A DSA consists of:

- Predefined legal background information (which can be derived following, e.g., the text of traditional legal contracts). A legal expert (e.g., in-house legal counsel) provides such description most of the times. This kind of data is unstructured by nature, that is data that are not organized in a predefined manner.
- Structured user-defined information, including the definition of the validity period, the parties participating in the agreement, the data covered and, most importantly, the statements that constrain how data can be shared among the parties (such statements usually include policy rules). Business policy experts and end users can be those who usually fill up this information and implement these fields.

For the aim of this paper, we define a DSA specification to be encapsulated (or wrapped) as an XML (eXtensible Markup Language) file. The XML format facilitates the task of programmatically accessing and working on the different DSA sections; furthermore, the XML fosters the interoperability with different tools and parties. The XML structure is described by an XSD (XML Schema Definition). An example will be given in the following.

From the analysis of many types of traditional legal data sharing contracts – and of some guidance issued by data protection authorities [7] – we identify the following sections to appear in an electronic DSA.

These are the examples of the *minimum essential* DSA sections:

- the *DSA title*, a label to identify the DSA into a repository of DSA.
- the *parties* involved into the DSA. For each party, we need to specify

  - its role in the DSA: borrowing the language from the privacy and data protection context, the DSA usually involves the Data Controller, the Data Processor and the Data Subject, terminology adopted in the European Parliament Directive 95/46/EC to indicate the parties involved in an agreement governing the sharing of personal data[1], and
  - its responsibility, i.e., the organisations duties which cannot be expressed in terms of authorisations and obligations by a data sharing rule, and for which the compliance checks cannot be enforced automatically (e.g., the role that each party will play in terms of gathering, sharing and storing the relevant data).
- the *validity* of the DSA: its start and end date, the duration of the any surviving obligations (especially, in relation to the use of data) after the expiration of the DSA and the duration of *off-line licences* for data access. The latter information allows the DSA actors to manage so called "off-line cases", as an example, when data are accessed by a mobile without Internet connection. This means that, in certain circumstances, data may be kept by the recipient also after the contract expires, for a predefined time.
- the *vocabulary* used for the DSA, which provides the terminology for authoring DSA statements. In our implementation, the vocabulary is defined by an ontology, written in OWL (Ontology Web Language) [1], that is a formal explicit description of a domain of interest. It provides the terminology for authoring DSA rules representing the semantics of terms in the context in which they are used and the relationships between them. Also, an ontology allows the reasoning and deductions in the scope of use. Such vocabularies are domain specific (e.g., medical context, legal context, etc.), but vocabularies describing cross-domain abstract aspects can be common for different context. The W3C (World Wide Web Consortium)[2] recommends some ontologies to describe objects and relationships across a number of domains. For instance, Org (The Organization Ontology)[3] is about organizational structures and the rules within them. FOAF (Friend of a Friend)[4] is one of the many available specifications about people and the relations between people and objects. The Platform for Privacy Preferences Project (P3P)[5] can be useful to express legal rules in different domains. The user can use a basic or proprietary vocabulary to describe rules about the parties or people involved in the agreement. However, in order to be more precise and specific in the DSA referring context, a domain specific ontology is more flexible. For example, the just mentioned Org may be suitable to the context of sharing data to and from mobile applications, moreover Core Person[6] can be very appropriated for the context of

[1] With a little abuse of notation, in this paper we use these terms also referring to other kind of data, to identify the actors involved in a general data sharing agreement.
[2] www.w3.org.
[3] www.w3.org/TR/vocab-org/.
[4] http://xmlns.com/foaf/spec/.
[5] http://www.w3.org/P3P/2004/040920_p3p-sw.html.
[6] https://joinup.ec.europa.eu/asset/core_person/description.

E-government and Public Administration, because it describes the fundamental characteristics of a person and it has already been used in public administrations contexts. Furthermore, the ontology SNOMED CT[7] is the most standardized terminology for health and it involves all radiological terms and procedures, thus it is very suitable to describe medical domains.

- the *data classification*, describing the nature of the data covered by the DSA. We consider two main data categories: personal data and non personal data. Additionally, we can propose a deeper data taxonomy for each of these classes, in order to identify better the object of the DSA. A (non-exhaustive) example follows:
  - Non personal data
    - Business data
      - Highly Confidential (e.g., strategic business plans, etc.)
      - Confidential (e.g., price lists, etc.)
      - Public (e.g., a list of products)
    - Administrative data (e.g., customers invoices, etc.)
  - Personal data
    - Common personal data
      - Identification details (e.g., name and surname)
      - Contact details (e.g., address, phone number)
      - ...
    - Special categories
      - Sensitive data (e.g., medical data)
      - Judicial data (e.g., data relating to offences or criminal convictions)
      - ...
- the *purpose* of the DSA, which is linked to the data classification; we assume that there is only one purpose for a DSA. If more than one purpose is needed, another agreement must be made. According to the data classification, the purpose can be:
  1. Administrative and Accounting (e.g., for booking, for payment);
  2. Healthcare services (e.g., for diagnoses);
  3. Scientific Research;
  4. Statistical (e.g., public costs control, epidemiological);
  5. Marketing (e.g., for commercial proposal of services/needs);
  6. Profiling (e.g., aggregation/grouping of users depending of certain user characteristics to propose specific products/services tailored to those characteristics);
  7. Fulfil law obligations (e.g., to access or share data in case of legitimate requests of public authorities).

It is worth noting that the Platform for Privacy Preferences Project (P3P) defines a long list of further purposes that could be considered.

We define also additional *technical* fields to support the DSA metadata:

- the *template id*; since a DSA can derive from another DSA in the three phases definition process (see the following Sect. 4), a template identifier can be useful to trace the original DSA.

[7] www.ihtsdo.org/snomed-ct/snomed-ct0.

– the *status*, that identifies the DSA into the three-steps process explained in Sect. 4 – possible values are: TEMPLATE, CUSTOMISED, and COMPLETED.

The following *optional* sections contain examples of the data sharing *rules* for a DSA:

– the *authorizations* section contains rules about permitted operations for each party (e.g., the possibility of sharing data with an identified third party);
– the *obligations* section contains rules about the duties of each of the parties in relation to the data sharing (e.g., the duty of not to transfer the data outside the country).

The Authorisation section contains a subsection specific for data subject rights, for example rights of viewing the data collected by the Data Controller, the source of data (where data has been obtained from, like from a public registry, etc.), cancellation, update, and their rights in front of third parties with whom the data have been shared (e.g., data can be stored, but cannot be accessed), grants/revokes, etc. Moreover, we distinguish among different types of obligations:

– Privacy: about personal data;
– Confidential: (usually) about business data;
– Audit: including obligations in relation to inspection of supervisory authorities (this may comprehend to specify logging actions related to data access);
– Warranties: concerning features, quality, and characteristics of the data: these obligations guarantee that the data are up to date, right, and complete (i.e., parties must share all the agreed data);
– Termination or expiration conditions for disposal of the DSA either for breach of contractual obligations, natural contract conclusion or convenience; they can include also what will happen to the data after the DSA disposal (i.e., delete or destroy data);
– Transfer of data: about only geographical movement of data (e.g., "not outside European Union");
– Other obligations (IPR, etc.).

It is worth noting that, even if we have defined Authorizations and Obligations to be optional sections, to have a significant DSA at least one authorisation – or one obligation – must be specified. Indeed, if both sections are empty, then the DSA does not explicitly impose any constraints.

While the required sections are not formal fields, but supporting metadata for the DSA, Authorizations and Obligations are authored using (semi)formal languages such as CNL4DSA [12] to define the data sharing rules, possibly using placeholders (strings), in case user preferences are needed for a statement (see next section for illustrative details of the use of placeholders).

Additional information can be provided by users:

1. Economics: the parties can specify fee for use of data, indemnities in case of breach in the use of data or penalties in case of improper sharing of data.

2. Governing Law: which law(s) the parties choose to apply to the business arrangements in the contract (i.e., commercial and jurisdiction clauses). In many countries, however, this field does not apply to the basic rules about processing personal data, because, according to many privacy and data protection laws, the identification of the applicable privacy and data protection law is based on mandatory criteria defined by the law.

An example of the XSD definition for DSAs defined according to the above sections is available online at www.iit.cnr.it/staff/marinella.petrocchi/dsa_schema.xsd.

4 DSA Authoring

The DSA Authoring Tool (hereafter also referred to as DSAAT) is a Web application for authoring DSAs. DSAAT allows organizations to define DSA, including the referring laws/regulations and acquiring end users' privacy preferences.

The actors involved in the DSAAT are:

A Legal Expert. S/He is very familiar with legal and contractual perspective content of agreement but s/he is not able to translate them in a high-level formal language which facilitates automatic processing of the policies. S/he is responsible for the creation of a DSA template, containing legal rules and optionally pending policies, that need to be completed by an end user. The use of a placeholder in a rule defines a pending policy.

A Policy Expert. S/He is responsible for defining business policies and other DSA metadata (as the ones listed in Sect. 3), specific to the context of a use case, starting from a DSA template. A business policy may require user preferences, so also a policy expert might include pending policies to be completed by the end user of a business application.

The End User. S/He may be involved when the DSA contains pending policies that require a user input to be finalized.

The process of authoring a DSA consists of three phases, each of them is managed by one of the above-cited actors though the DSAAT. The DSA can have the following status:

1. TEMPLATE: the legal expert creates the DSA template, i.e., a draft version of the DSA, containing the legal policies. It can be reused between different business use cases.
2. CUSTOMISED: the policy expert populates the DSA template with business policies, specific for the context of a use case. The DSA moves from template status to customised status. A customised DSA might still contain policy placeholders, used to gather a end user preference. Pending policies take the form of check-boxes (e.g., "allow consent to use data for marketing purposes. (Yes, No)") or free text fields (e.g., "delegate access to (XYZ) person").
3. COMPLETED: if there are no rules that require specific choices of the end user, or after the user preferences gathering, the DSA can be considered completed.

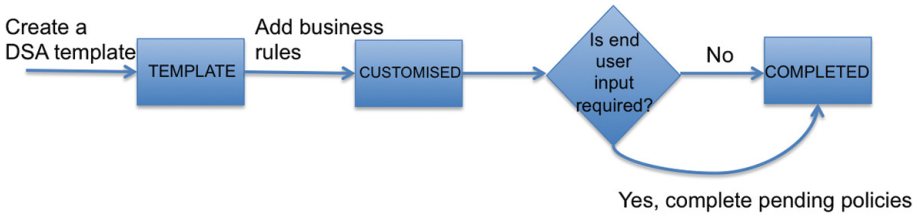

Fig. 2. The DSA state-chart diagram

Figure 2 shows a state diagram of DSA authoring in the DSAAT context.

DSA Authoring Tool Storyboard. The DSAAT home page shows the content of a DSA repository (see Fig. 3). For each entry in the repository the interface shows the UUID (Universal Unique Identifier), which is the name of the DSA file (in XML format), the size of the file, the title of the DSA and its status. The figure shows that user is logged as legal expert, meaning that s/he can either create a new DSA template, or edit and view an existing one.

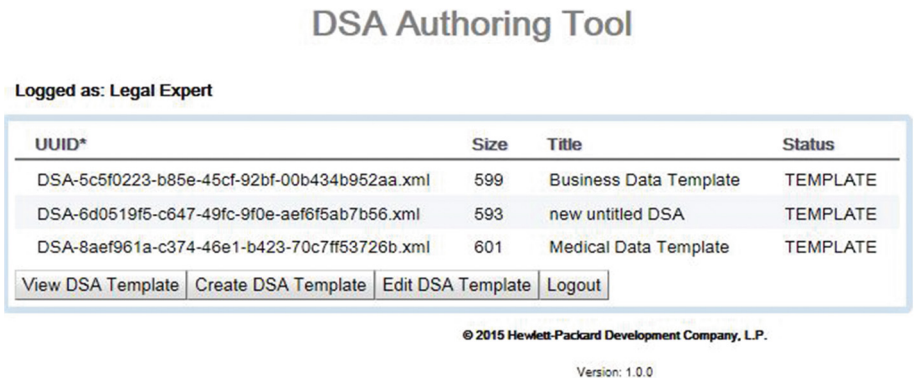

Fig. 3. DSAAT home page interface

The following gives details on the actions which the DSA actor can perform via the DSAAT interface:

1. create a new DSA template - Typically a legal expert creates a new DSA template. First of all, a reference to the vocabulary, for the definition of terms and actions used by the policies, is required. In DSAAT, different ways of loading a vocabulary (technically, a file written in OWL - Ontology Web Language) can be supported: a user can provide a URL, so that the DSAAT can access it through HTTP or a path from his/her file system, if the vocabulary is stored locally. The DSAAT takes care of fetching and processing the ontology defining a vocabulary, and of using it when the user edits a DSA.

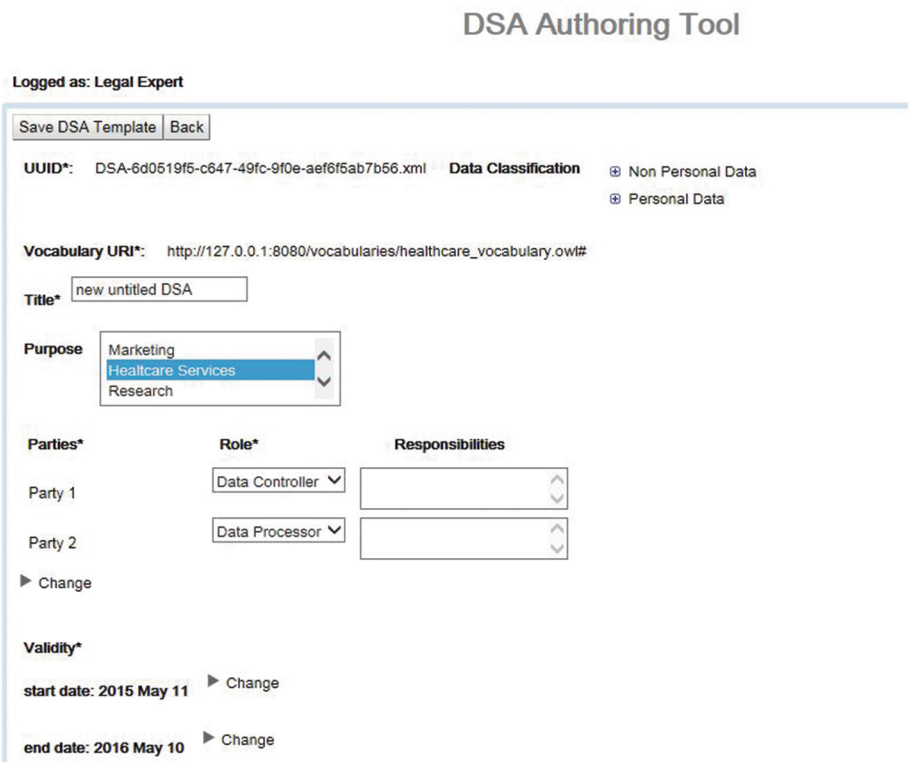

Fig. 4. DSA (template) creation

Once a vocabulary has been selected, the web page shows to the user an input form with information to be filled, describing the DSA.
Thus, the user must specify for a DSA according to the structure described in Sect. 3 (XML container) and through the interface illustrated in Fig. 4:

- a DSA *title*;
- the *parties* involved into the DSA. S/he can select the parties from a drop-down list menu. For each party, the user must specify a role in the DSA, selecting one item from another menu and, optionally, specifying the party's responsibility in the agreement. The responsibilities are hosted in a free-text field because no enforcement will be provided for this information. It is worth noting that the figure shows only the role of the parties defined. Indeed, being the form at DSA template level, the specific name of the parties remains generic. The policy expert at organization level will be responsible for fill them in.
- the *validity* of the DSA; its start and end date, and the duration, in days, of off-line licences for the data access. This field may be refined by the policy expert at the organization level.

- the *data classification*; note that to support the taxonomy of the data classification, once the legal expert selects the kind of data, the user interface allows to go in depth with a drop down menu, as shown in Fig. 5, where business data, which in their turn belong to the "non personal data" category, are expanded into "highly confidential", "confidential", and "public";
- the *purpose* of the DSA, which is strictly connected with the data classification. The legal expert can select items in the menu containing the possible value, according to the XML schema (see details in Sect. 3);
- at least one statement in either *Authorization* or *Obligation* sections of the DSA, as defined into the schema. We remind the reader that legal experts are supposed to encode at DSA template level terms of law that apply for the purpose of sharing data belonging to a certain class. In the advanced phase of authoring, the policy expert and - optionally - the end user will append organisation-specific data sharing rules and end user privacy preferences.

Data sharing rules include both authorisations and obligations. In the following, we will focus on authorisations, even if the same reasoning hold for obligations. Thus, when the generic user is going to add a new authorisation, a pop-up will be displayed to ask if s/he is going to define a data subject authorisation, so that the tool can put the rule in a new separated section, as Fig. 6 shows.

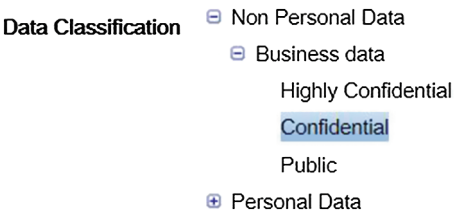

Fig. 5. DSA data classification (excerpt)

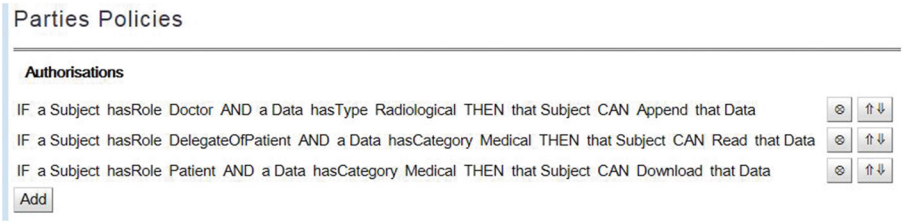

Fig. 6. A detail of the authorisations section

A dedicated section about sharing the data with other organisations exists in the DSA: it contains the definition of policies where third-parties are involved.

It describes if and how the receiving party of the DSA is permitted (or not) to share the data with any third party and any relevant restriction.

Each definable rule is expressed in terms of authorisations and obligations. The user is continuously supported in the editing of the various sections of a DSA, especially in the definition of the rules, in an intuitive and assisted way. The DSAAT provides suggestions in the definition of the rules according to the initially chosen vocabulary. For instance, if the user is writing a rule about a certain entity in the vocabulary, the user interface provides a pop-up containing only the predicates and then the objects for which a reference in the ontology defining the vocabulary exists. The definition of the rules is error free (from a semantic point of view): this approach allows the user to insert only well-formed rules according to the reference ontology. Figure 7 shows an example of the kind of suggestions shown to the user when editing the Authorisations section.

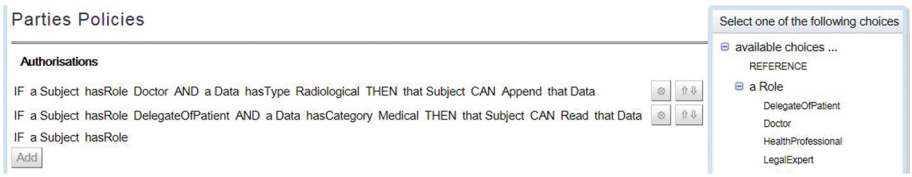

Fig. 7. Suggestions for authorisations completion

It is worth noting that rules may contain a placeholder to acquire the end user input in the third – and last – phase of the DSA definition process described in Sect. 1. The use of a placeholder in a rule defines a pending policy.

2. Create a new DSA starting from a template. Once the legal expert provides the first round of authoring, the DSA is in TEMPLATE status. It means that this DSA can be loaded and used as a starting point by a policy expert to create a custom copy of the DSA, according to the use case.
   Organisations may establish specific DSA with other parties starting from a catalogue of templates: a policy expert creates a new DSA starting from an existing DSA template in the catalogue. The policy expert adds new business policies and populates some sections of the DSA according to the organizational context in which it operates. The policy domain expert uses the same DSA editor used by legal expert, but since s/he wants to create a new DSA, s/he can save a new copy of the DSA, thus all the changes are not impacted on the original DSA template. Once the business policies have been defined, DSA can be finalized (with all the metadata): if there is no need to complete pending policies by end users, the DSA moves directly to the COMPLETED status. Otherwise, a business application loads the DSA and presents it to the end user to get his/her privacy preferences.
3. Edit an existing DSA. All kinds of DSA editing actors can modify a DSA or a DSA template stored into the DSA repository during any of the phases of

the authoring process. The DSA can remain in the same status it was stored or pass to the following phase according to the user actions.

4. View an existing DSA. As mentioned before, the home page of the DSAAT shows the content of the DSA repository. From this view, a user can select an existing DSA and view its content in a XML format.

5 Concluding Remarks

In this paper, we have proposed fields (and format) of an electronic data sharing agreement following guidelines for traditional paper contracts. The creation of the agreement is supported by an authoring tool for the definition of the contractual clauses as well as appropriate metadata rendering the original paper document.

The authoring phase depicted above is a three-step process, where first legal experts define terms of Law and regulations applicable according to the data classification and the purpose of data sharing; then, policy experts at organization level define specific business policies. Finally, end users may insert preferences for sharing data referring to them. A well formed DSA implies that the rules defined at the three levels of authoring do not conflict one with each other. A conflict may arise when two applicable rules deny and allow, at the same time, the access to the same data, by the same subject, under the same contextual conditions. Thus, a sound management of DSA should involve the support of a policy conflict analyser, detecting conflicts between rules edited at DSA template level and organisation level, and between rules edited at organisation level and end users' preferences. An example of conflict analyser is in [13], which we are currently adapting to be usable in our framework. Once a conflict is detected, we also envisage to have an automatic conflict resolution procedure that chooses, among conflicting policies the "right" one, to be enforced. Our research effort turns around technical compliance with terms of Law. Thus, an appropriate strategy for conflict resolution could be, e.g., the "Lex superior derogat legi inferiori" Roman law principle, meaning that the higher ranking legal source overrides the lower ranking one. Furthermore, other strategies could be exploited as well (like the principle "Lex specialis derogat legi generali", a complementary principle meaning that exceptions may override a more general regulation). In past work, we developed a prototypal conflict solver that can be easily adapted to prioritise data sharing rules according to the authoring level at which they have been edited [10,11]. Evolution of the conflict analyser and adaption of conflict solving strategies are left for future work. Finally, the direct involvement of end users in the specification of policies expressing privacy constraints paves the way for usability studies on the proposed three-step authoring phase. We are currently investigating how individuals can be actively and easily involved in specifying their preferences. As a running scenario, we consider the constrained access to patients' radiological examinations being stored at a Cloud provider.

References

1. Antoniou, G., Harmelen, F.V.: Web ontology language: OWL. In: Staab, S., Studer, R. (eds.) Handbook on Ontologies in Information Systems, pp. 67–92. Springer, Heidelberg (2003)
2. Axiomatics. www.axiomatics.com. Accessed 22 December 2015
3. Brodie, C., et al.: An empirical study of natural language parsing of privacy policy rules using the SPARCLE policy workbench. In: SOUPS. ACM (2006)
4. Brodie, C., et al.: The coalition policy management portal for policy authoring, verification, and deployment. In: POLICY, pp. 247–249 (2008)
5. Casassa Mont, M., Matteucci, I., Petrocchi, M., Sbodio, M.: Towards safer information sharing in the cloud. Int. J. Inf. Secur. **14**, 319–334 (2015)
6. Consequence Project. Infrastructure for data sharing agreements, December 2010. http://goo.gl/is7cpR
7. Information Commissioner's Office (ICO). Data sharing code of practice, pp. 26, 41–45 (2011). https://goo.gl/11vXHb. Accessed 22 December 2015
8. Johnson, M., Karat, J., Karat, C.-M., Grueneberg, K.: Optimizing a policy authoring framework for security and privacy policies. In: SOUPS, pp. 8:1–8:9. ACM (2010)
9. Karat, J., et al.: Designing natural language and structured entry methods for privacy policy authoring. In: Costabile, M.F., Paternó, F. (eds.) INTERACT 2005. LNCS, vol. 3585, pp. 671–684. Springer, Heidelberg (2005)
10. Lunardelli, A., Matteucci, I., Mori, P., Petrocchi, M.: A prototype for solving conflicts in XACML-based e-Health policies. In: Computer-Based Medical Systems, pp. 449–452. IEEE (2013)
11. Matteucci, I., Mori, P., Petrocchi, M.: Prioritized execution of privacy policies. In: Di Pietro, R., Herranz, J., Damiani, E., State, R. (eds.) DPM 2012 and SETOP 2012. LNCS, vol. 7731, pp. 133–145. Springer, Heidelberg (2013)
12. Matteucci, I., Petrocchi, M., Sbodio, M.L.: CNL4DSA: a controlled natural language for data sharing agreements. In: SAC: Privacy on the Web Track, pp. 616–620. ACM (2010)
13. Matteucci, I., Petrocchi, M., Sbodio, M.L., Wiegand, L.: A design phase for data sharing agreements. In: Garcia-Alfaro, J., Navarro-Arribas, G., Cuppens-Boulahia, N., de Capitani di Vimercati, S. (eds.) DPM 2011 and SETOP 2011. LNCS, vol. 7122, pp. 25–41. Springer, Heidelberg (2012)
14. OASIS. eXtensible Access Control Markup Language (XACML) version 3.0, January 2013
15. Reeder, R.W., Karat, C.-M., Karat, J., Brodie, C.: Usability challenges in security and privacy policy-authoring interfaces. In: Baranauskas, C., Abascal, J., Barbosa, S.D.J. (eds.) INTERACT 2007. LNCS, vol. 4663, pp. 141–155. Springer, Heidelberg (2007)
16. Rosenthal, S.S.: Specifying data sharing agreements. In: Seventh IEEE International Workshop on Policies for Distributed Systems and Networks, pp. 157–162 (2006)
17. Swede, S.: Enforcing scientific data sharing agreements. In: IEEE 9th International Conference on e-Science, pp. 271–278 (2011)
18. Wishart, R., Corapi, D., Marinovic, S., Sloman, M.: Collaborative privacy policy authoring in a social networking context. In: POLICY, pp. 1–8. IEEE (2010)