Semantic Model for Legal Resources: Annotation and Reasoning over Normative Provisions

Editor(s): Pompeu Casanovas, Universitat Autònoma de Barcelona, Spain, and Victoria University, Australia; Monica Palmirani, University of Bologna – CIRSFID, Italy; Silvio Peroni, University of Bologna, Italy, and National Research Council, Italy; Tom van Engers, University of Amsterdam, The Netherlands; Fabio Vitali, University of Bologna, Italy
Solicited review(s): Eva Blomqvist, LinkÃűping University, Sweden; Paolo Ciccarese, Harvard Univeristy, USA, and Massachusetts General Hospital, USA; JosÃľ Manuel LÃşpez-Cobo, playence, Spain

Enrico Francesconi
ITTIG-CNR, via de' Barucci 20, Florence (Italy)

Abstract. A Semantic Web approach for an advanced access to legislative documents is presented in terms of a model of normative provisions and related axioms. In particular, relations between provisions are identified and modeled by introducing patterns able to describe Hohfeldian legal fundamental relations. Moreover, a query-based approach able to deal with relations between provision specific instances is described. Examples of semantic annotation of legal textual resources using RDF/OWL standards, as well as advanced access and reasoning facilities over provisions using SPARQL, are shown. The main benefit of the approach is represented by the ability to keep the complexity of the problem within a description logic computational tractability.

Keywords: Legal Semantic Web, Normative Provisions, Provision Relations, Hohfeldian Reasoning, Description Logic, OWL-DL

1. Introduction

The legal domain is one of the most challenging areas for developing applications based on the Semantic Web principles, because of the complex nature of legal information and document workflow, as well as the peculiarities of legal users' information needs, which require advanced information retrieval and reasoning services. As regards legislation, in particular, users are mainly interested in accessing norms rather than simply documents; they are particularly interested in knowing the relations between norms, having support to legal reasoning and consultancy services, as well as instruments to check procedures compliance with respect to specific statutes and regulations.

The development of advanced retrieval and reasoning services over norms can benefit from the description of the legislative texts semantics at different granularity levels: at the level of terms, thus describing the concepts actually expressed; at the level of the whole document, thus providing information on the subject matter of the act; at the specific level of norms, thus identifying duties, rights, sanctions, permissions, procedures, etc., such documents may contain, as well as actors and actions involved. Such semantic description allows legal practitioners and citizens to retrieve not only documents concerning a particular domain (for example the consumer law domain), but also the norms addressed to a specific actor or about specific actions.

Let's consider a user who is willing to subscribe to a "distance contract", for example through the Internet, in order to buy a financial service: the user might be interested to know which are his rights (as for example the right of withdrawal), either explicitly expressed or implicitly inferred because expressed in the form of a duty of the supplier towards the consumer, which are actually rights of the consumer himself. In this respect it would be useful to have a system able to retrieve the specific portions of legislation reporting the norms of interest, and also able to infer the norms that are implicitly expressed concerning the same type of rules. A similar scenario is the one in which a local public administration has to draft a Regulatory Impact Analysis (RIA) document. In this case the analysis of the impact on the legal order of the proposed norms, from the point of view of their compliance with the Constitution and the powers of the local authorities, will benefit from an advanced access and reasoning system on legislation.

One of the pre-conditions for implementing such services is to rely on legislative texts properly marked-up according to structural and semantic models for the Law. In literature several models (classification) of legal norms have been proposed, from the traditional Hohfeldian theory of legal concepts [20] until more recent legal philosophy theories [23] [17] [25] [5] [21], while related computational models have been implemented [19] [16]. Nevertheless such computational models deal with ontologies and rules, whose combination is usually undecidable [18], without addressing the problem of identifying the reasoning schemas over norms that can be managed within a Description Logic (DL) computational complexity. Reasoning over norms within a DL framework guarantees the computational tractability of the problem and the possibility to rely on available automatic reasoners, without the need to implement tailored ones.

In this paper an approach for the development of advanced access and reasoning facilities on legislation within a DL computational tractability is presented. It is based on a definition of a semantic model for legislation in terms of normative provisions, presented in Section 2, which can be used to provide semantic description refinement to legislative resources available at a minimum level of structural mark-up. In particular, according to a view proposed in [6] [7] specifically anchored to the structure of legislative texts, laws and regulations may be seen as a set of *provisions*, carried by speech acts [28] [24]. Following this perspective, fragments of a legislative text are, at the same time, sentences, paragraphs, or provisions, according to whether they are seen from a formal or semantic view-point. In this context, in Section 3 possible kinds of relations between provisions are described. In Sections 4 an extension of the normative provisions model through Description Logic patterns able to deal with relations between provisions, as the Hohfeldian fundamental relations [20], are presented. In Section 5 an example of how this approach can support Hohfeldian inferences for improving provisions accessibility within a Description Logic framework is presented with respect to a European directive excerpt. In Section 6 specific relations which can be identified between provision instances are discussed, while in Section 7 an implementation of reasoning facilities over provision instances with respect to the same European directive excerpt is shown. Finally, in Section 8, some conclusions are reported.

2. A model of normative provisions

According to the model of normative provisions presented at first in [6] and [7], provisions can be described in terms of *provision types* (as *Term Definition*, *Procedure*, *Duty*, *Right*, *Power*, as well as more technical ones as *Insertion*, *Abrogation*, *Substitution*, etc.) and related *attributes*[1] (for example the *Bearer* of a *Right*, or the *Definiendum* of a *Term Definition*), reflecting the lawmaker directions. Provision types and attributes can be considered as a sort of metadata model able to analytically describe fragments of legislative texts, hence the name of *Provision Model* [6].

The details of the Provision Model are widely described in [6] and [7]; in this paragraph the semantic organization of the model is briefly recalled.

In the Provision Model, provision types are organised into two main groups: *Rules* (introducing entities or expressing deontic concepts) and *Rules on Rules* (different kinds of amendments). Adopting a typical law theory distinction, well expressed by Rawls [23], *Rules* consist in:

- *Constitutive rules*, which introduce or assign a juridical profiles to entities of a regulated reality;
- *Regulative rules*, which discipline actions or their substantial and procedural defaults (remedies).

On the other hand, *Rules on Rules* can be distinguished in:

- *Content amendments*, which modify literally the content of a norm, or their meaning without literal changes;
- *Temporal amendments*, which modify the times of a norm (come-into-force and efficacy times);
- *Extension amendments*, which extend or reduce the cases on which the norm operates.

[1] also called *arguments* in [6]

The values of *provision attributes* can be expressed by lexical units, or by concepts derived from thesauri/ontologies, able to provide additional information on the entities of the regulated domain [3] [19].

For example, the following fragment (article 5, paragraph 1) of the European Directive 2002/65/EC, concerning the distance marketing of consumer financial services:

> *The supplier shall communicate to the consumer all the contractual terms and conditions and the information referred to in Article 3(1) and Article 4 on paper or on another durable medium available and accessible to the consumer in good time before the consumer is bound by any distance contract or offer.*

besides being considered as a formal partition (a *paragraph*) of the related directive, can also be viewed as a semantic component (a *provision*) and qualified as a *Duty*, whose attributes, expressed in attribute-value pair notation, are:

hasBearer =	'Supplier'
hasObject =	'Contractual terms and conditions ...'
hasAction =	'Communication'
hasCounterpart =	'Consumer'

where attributes values can be literals or concepts in an ontology.

An example of ontology for the European consumer law has been developed within the DALOS project[2] [1]. In this paper concepts described in the DALOS ontology[3] will be used for describing related concepts expressed in the Directive 2002/65/EC, which will be used as an example to illustrate the approach.

3. Relations between provisions

Relations between provisions can be identified in order to highlight the meaningful links between different types of norms and to pave the way for reasoning over norms, expanding information actually selected by a norms retrieval system. Two kinds of relations between provisions can be identified: *logical relations* and *technical relations*.

Logical relations are relations between provisions which are necessary from a logical point of view, as the classical Hohfeldian relations. Hohfeld [20] identifies two relational schemes between provisions. The first logical relations scheme involves deontic concepts in terms of correlative relations between Right and Duty, as well as No-right and Privilege, and opposite relations between Right and No-right, as well as Duty and Privilege (Fig. 1).

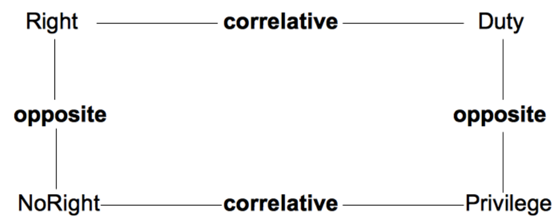

Fig. 1. Deontic concepts and their relations.

For example, if A has a right towards B, this is equivalent to B having a duty towards A. Similarly, if B has a privilege towards A, which means that B can do whatever he or she wants because B has no duty to refrain from doing it, A has no right to prohibit B from doing so.

The second logical relations scheme involves potestative[4] concepts, in terms of correlative relations between Power and Liability, as well as Disability and Immunity, and opposite relations between Power and Disability, as well as Liability and Immunity (Fig. 2).

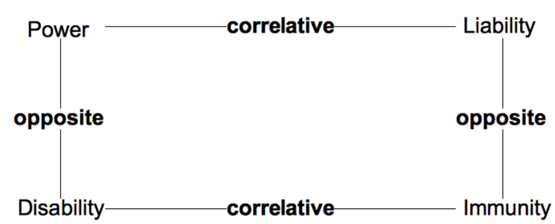

Fig. 2. Potestative (anakastic) concepts and their relations.

For example if A has a power towards B, this is equivalent to B having a liability (namely a subjection) with respect to A. Similarly, if B has an immunity with respect to a liability (subjection to a power of A), it means that A is disabled to limit B's immunity.

Technical relations between provisions, on the other hand, are relations not necessary from a logical point of view, but they derive from legislative techniques considerations; this means that they are possible and can be identified in legislative texts provided that the legislative drafter follows specific legislative technique recommendations in expressing such provisions. An example of such relations is the one existing between a *Term Definition*, introducing a concept identified by the attribute *Definiendum*, and all the other provisions having, as an attribute value, the value of such

[2] http://www.dalosproject.eu

[3] http://godel.ittig.cnr.it/ontologies/ConsumerLaw.owl

[4] also called 'anankastic in [11], expressing necessity conditions

Definiendum. Another example can be the relations between the *Duty* of a specific *Bearer* to accomplish a specific *Action* towards a given *Counterpart*, as well as the *Procedure* to fulfill it.

While *logical relations* can be described at the level of Provision Model (see Sections 4-5) and are inherited by the related instances, *technical relations* can be identified and described at the level of provision instances only, because linked to the provision instances content (see Sections 6-7). As reported in [7] technical relations between provisions can be established directly by the legislator through references, within or out of the including act, or can be deduced by reasoning over provisions content, expressed by provisions attribute values. The relations established by the legislator through references can be easily detected and assume specific roles especially in handling amendments, in particular as regards automatisms that can be conceived to produce consolidated versions of legislative texts, as well as a guide in their consultation. The relations that can be deduced by reasoning, either logical or technical ones, are obviously more difficult to identify. This difficulty represents one of the main sources of ambiguities in the application of the law, therefore they will be discussed in Sections 6 and 7.

4. Logical relations between provisions

Logical relations between provisions can be described as axioms on provision types and attributes, as the Hohfeldian fundamental relations regarding *Right/Duty*, *Liberty/No-right*, *Power/Liability*, *Immunity/Disability*, as well as the relation between the *Duty* of a subject (duty *Bearer*) towards a *Counterpart*, which can be viewed as an implicit *Right* of the duty *Counterpart* towards the duty *Bearer*.

As previously discussed, a description of legislative texts in terms of provisions allows advanced access services on legislation, able to give reasoning facilities based on the theory of norms. A typical example can be a service able to exploit the previously mentioned logical relations by accessing the rights of a subject, either explicitly expressed or inferred. This can be obtained by describing the logical relations between *Duty* and *Right* at the level of the Provision Model.

For example article 5 paragraph 1 of the European Directive 2002/65/EC reported in Section 2, can be considered a provision of type *Duty* involving 'Supplier' and 'Consumer'. In terms of Provision Model, such Duty of the 'Supplier' towards the 'Consumer' can be expressed in functional notation[5] [18] as follows:

Duty(hasBearer='Supplier', hasCounterpart='Consumer')

which corresponds to

Right(hasBearer='Consumer', hasCounterpart='Supplier').

This Hohfeldian relation underlines an equivalence between *Duty* and *Right*, representing the logical correlation between them, as long as the values of the duty *Bearer* and *Counterpart* are swapped, assuming symmetric roles in the *Right* provision, therefore involving equivalence relations between provision types and attributes. However, describing these relations in the Provision Model by establishing the equivalence relations *Duty* $\equiv$ *Right*, as in [9], and *hasBearer* $\equiv$ *hasCounterpart* would imply equivalence relations between *any* duties and rights, irrespective to the attribute types and values, as well as between all the provision types sharing equivalence relations between such attributes, which might produce inconsistent results in a provisions retrieval system.

For example a query aiming to retrieve provisions having *Right(hasBearer = 'Supplier')*, would also give back *Duty* provisions having *Duty(hasBearer = 'Supplier')* because they satisfy the axiom *Duty* $\equiv$ *Right*. Similarly, the previously mentioned query would retrieve back *Right* provisions having *Right(hasCounterpart = 'Supplier')*, since they satisfy the axiom *hasBearer* $\equiv$ *hasCounterpart*.

To avoid these problems, while relying on Description Logic[6] expressivity as implemented in OWL-DL[7], an extension of the Provision Model, described in OWL [18], is proposed.

4.1. Extension of the Provision Model

Firstly provision attributes are specified according to the related provision types, for example *hasBearer* and *hasCounterpart* attributes are distinguished in terms of hasDutyBearer and hasDutyCounterpart as properties of Duty, and hasRightBearer and hasRightCounterpart as properties of Right.[8]

[5] in this notation provision types are binary functions between provision attribute-value pairs as functional variables

[6] decidable subset of the First-Order Logic

[7] decidable specialization of Ontology Web Language (OWL) implementing the Description Logic

[8] Hereinafter, provision types as OWL classes (starting with capital letters) and provision attributes as OWL properties (starting with

A model extension at the level of provisions type can also be provided by observing that a *Right*, in correlative correspondence with a *Duty*, is actually not explicitly expressed in the text, but represents an implicit provision, basically a different view of the *Duty* itself, where the values of the related bearer and counterpart attributes are swapped. Therefore, the Provision Model can be extended in terms of Duty and Right implicit and explicit disjoint subclasses, able to represent a complete covering of the related superclass (ex: ExplicitRight and ImplicitRight disjoint subclasses represent a complete covering of the Right superclass).

Attributes can also be specified as regards both implicit and explicit provisions, so that hasImplicitDutyBearer and hasExplicitDutyBearer are sub-properties of hasDutyBearer, as well as hasImplicitRightBearer and hasExplicitRightBearer are sub-properties of hasRightBearer (see Fig. 3 for the extension of the Right provision type and attributes; a similar extension can be figured out for Duty and other provisions).

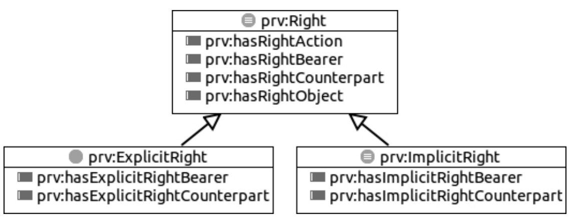

Fig. 3. Extended Right attributes.

For each attribute (property) both *domain* and *range* can be specified: *domain* specifies the type of individuals a provision attribute applies to (e.g. the individuals of the class ExplicitDuty for a provision attribute hasExplicitDutyBearer); *range* specifies the type of values of this provision attribute. Since legislative texts can deal with any aspects of the reality, the values of provision attributes which are modeled by OWL object properties may belong to any class of objects (individual of owl:Class). Note that this cannot be formally defined in the ontology in order to guarantee that the ontology is fully-compliant to the OWL 2 DL profile.

Note that only explicit provision classes (and consequently explicit properties) will be used to mark-up textual provisions, as they are the only provisions actually (explicitly) expressed in legislative texts, while implicit provision classes act as a sort of “abstract” classes, which will be used for reasoning.

lowercase letters) are written in serif font. The namespace is omitted for simplicity.

4.2. Hohfeldian relations in the Provision Model

As an example of logical relations implementation in the Provision Model, let’s consider the following Hohfeldian relations:

– the couple *Duty/Right* as examples of correlative deontic concepts;
– the couple *Power/Liability* as examples of correlative potestative concepts[9] [26].

Similar considerations can be given for the deontic couple *Liberty/No-right* and the correlative potestative one *Disability/Immunity*, because they can be derived as negation of the opposite deontic and potestative couples, respectively.

To represent the Hohfeldian fundamental relations between Duty and Right, firstly an equivalence relation between their explicit and implicit views is established: ImplicitRight $\equiv$ ExplicitDuty and ImplicitDuty $\equiv$ ExplicitRight. In Fig. 4 the established sub-class (Section 4.1) and equivalence relations between Duty and Right in their explicit and implicit views are summed up.

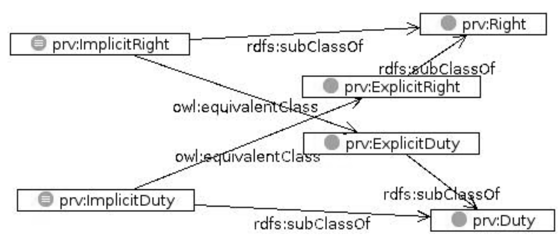

Fig. 4. Sub-class and asserted equivalence relations between Duty/Right deontic correlative provisions.

As for the couple *Duty/Right*, an equivalence between *Power* and *Liability* represents an Hohfeldian relation as long as the values of *Bearer* and *Counterpart* of a *Power* are swapped, assuming symmetric roles in the *Liability* provision, therefore involving, in the Provision Model, equivalence relations between provision types and attributes. Such group of relations is isomorphic to the one between Duty and Right, therefore the same relational pattern can be applied to represent Power/Liability relations, as summed up in Fig. 5.

Moreover, equivalence relations between implicit/explicit Duty and Right attributes, as well as between implicit/explicit Power and Liability attributes, can be established. In Fig. 6 the asserted sub-property and equivalence relations between hasDutyBearer and hasRightCounterpart in their explicit and implicit views

[9] also called ‘anankastic’ in [11], expressing necessity conditions

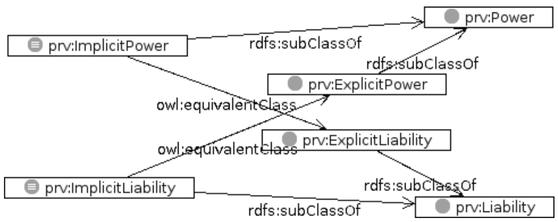

Fig. 5. Sub-class and asserted equivalence relations between Power/Liability potestative correlative provisions.

are shown. The same holds for the asserted sub-property and equivalence relations between hasPowerBearer and hasLiabilityCounterpart in their explicit and implicit views.

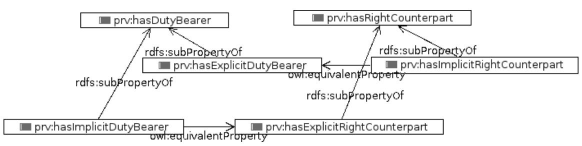

Fig. 6. Asserted sub-property and equivalence relations between hasDutyBearer and hasRightCounterpart in their explicit and implicit views.

The reader can imagine a symmetric view for the relations between a right bearer and a duty counterpart, as well as between a liability bearer and a power counterpart in their explicit and implicit views.[10]

Note that the proposed patterns do not interfere with the equivalence relation between Right and Duty, as well as Power and Liability, which still hold. In fact, for the couple Right/Duty as example (but similar consideration can be given for Power/Liability), an individual of ExplicitDuty is also an individual of Duty, given the axiom rdfs:subClassOf(ExplicitDuty, Duty). Moreover the axiom owl:equivalentClass(ImplicitRight, ExplicitDuty) tells us that such individual is also an ImplicitRight, which is also a Right, given the axiom rdfs:subClassOf(ImplicitRight, Right). Since this is done symmetrically for explicit and implicit duties and rights, we can deduce that Right is equivalent to Duty, given that the union of the disjoint explicit and implicit subclasses covers completely the related superclass (see Section 4.1).

Therefore provisions properties are preserved, but the expressivity of the model is improved to provide enhanced retrieval and reasoning services. The proposed pattern in fact aims to introduce:

[10]A first OWL 2 DL release of the Provision Model, limited to demonstrate this approach is available at http://godel.ittig.cnr.it/ontologies/ProvisionModel.owl

1. Properties equivalence, allowing direct swapping on attributes contents for addressing provision relations, without the need of using conditional statements (ex: if the value of hasDutyCounterpart is ‘Consumer’ then ...)
2. Abstract classes (namely classes not used for mark-up, in our case “implicit” classes) so to provide different views (implicit and explicit views) on the same provision, as well as retrieval services able to access implicit provisions only (ex: provision instances where ImplicitRightBearer is ‘Consumer’).

Moreover, by providing equivalence relations between symmetric implicit and explicit classes and attributes, the proposed pattern is able to avoid inconsistent deductions (as for example that bearers and counterparts freely mix in the same provision), producing on the other hand inferential deductions (for instance attribute mixing) which keep semantic consistency. For example, given the following explicit right:
a) *ExplicitRight(hasExplicitRightBearer = ‘Consumer’)*
given that:
ExplicitRight $\equiv$ *ImplicitDuty*
hasExplicitRightBearer $\equiv$ *hasImplicitDutyCounterpart*
the following consistent deductions, describing the same provision instance, can be obtained:
b) *ExplicitRight(hasImplicitDutyCounterpart = ‘Consumer’)*
c) *ImplicitDuty(hasExplicitRightBearer = ‘Consumer’)*
d) *ImplicitDuty(hasImplicitDutyCounterpart = ‘Consumer’)*
which are semantically consistent. For example, being “Consumer” an explicit bearer of an explicit *Right*, it is also to be considered an implicit counterpart of the same provision, viewed as an implicit *Duty*.
Finally, it is worth to stress that the introduced axioms are not dealing with relations between different provision instances expressed in a legislative text (which could be better described in terms of existential restrictions, as for example: ‘for every explicit duty there is an implicit right where bearer and counterpart are swapped’, or in terms of relations between attribute values (Section 6)), but they deal with different views (explicit and implicit views) of the *same* provision instance. In this perspective all the deductions derived from the established equivalence relations between classes, as well as the deductions derived from mixing provision qualified properties, are valid, as previously discussed.

5. Logical relations between provisions example

In this section an example of how this approach can be used for a provision retrieval system able to deal with logical relations is shown. In particular an example of Hohfeldian reasoning over provisions is described.

5.1. Semantic annotation

Let's first consider an excerpt of Directive 2002/65/EC[11], properly annotated using a CEN-Metalex [10] compliant mark-up syntax (here below), where articles, paragraphs, sub-paragraphs and inline relevant textual fragments (<span>) are marked-up and identified by using specific IDs which follow an established convention[12]. Such mark-up is provided since, according to [7], provisions are typically represented by paragraphs of legislative texts (in the example marked-up by the element `<paragraph>`) while provision attributes can be explicitly expressed in fragments of paragraphs (in the example marked-up by `<span>`) or not explicitly expressed.

```
<article id="art2">
 <paragraph id="art2;par1">
For the purposes of this Directive:
[...]
 <subparagraph id="art2;par1;subc">
  (c) "supplier" means <span id="art2;par1;subc;spa1">any
  natural or legal person, public or private, who, acting in
  his commercial or professional capacity, is the contractual
  provider of services subject to distance contracts;</span>
 </subparagraph>
 <subparagraph id="art2;par1;subd">
 (d) "consumer" means <span id="art2;par1;subd;spa1">any
 natural person who, in distance contracts covered by this
 Directive, is acting for purposes which are outside his trade,
 business or profession;</span>
 </subparagraph>
</paragraph>
[...]
</article>
[...]
<article id="art5">
 <paragraph id="art5;par1">
   1. The supplier shall communicate to the consumer all the
   contractual terms and conditions [...]
 </paragraph>
 <paragraph id="art5;par2">
   2. The supplier shall fulfil his obligation under paragraph 1
   immediately after the conclusion of the contract [...]
 </paragraph>
 <paragraph id="art5;par3">
   3. At any time during the contractual relationship the
   consumer is entitled, at his request, to receive the
   contractual terms and conditions on paper.[...]
 </paragraph>
</article>
<article id="art6">
 <paragraph id="art6;par1">
   1. The Member States shall ensure that the consumer shall
   have a period of 14 calendar days to withdraw from the
   contract without penalty [...]
 </paragraph>
 [...]
</article>
<article id="art7">
 [...]
 <paragraph  id="art7;par2">
   2. Member States may provide that the consumer cannot be
   required to pay any amount when withdrawing from an
   insurance contract.
 </paragraph>
 </article>
 [...]
<article id="art11">
 <paragraph id="art11;par1">
   Member States shall provide for appropriate sanctions in
   the event of the supplier's failure to comply with
   national provisions adopted pursuant to this Directive.
   They may provide for this purpose in particular that the
   consumer may cancel the contract at any time, free of
   charge and without penalty. [...]
 </paragraph>
</article>
```

According to the Provision Model and a domain ontology like DALOS, the semantics of such document fragments, identified by the including document URI and specific IDs, can be summed up as in Tab. 1 (this semantic description is limited to the attributes useful to demonstrate the approach).

Partition ID	Provison Type	Provision Attributes
art2;par1;subc	TermDefinition	hasDefiniendum='Supplier' hasDefiniens="#art2;par1;subc;spa1"
art2;par1;subd	TermDefinition	hasDefiniendum='Consumer' hasDefiniens="#art2;par1;subd;spa1"
art5;par1	ExplicitDuty	hasExplicitDutyBearer='Supplier' hasExplicitDutyAction='Communication' hasExplicitDutyObject='ContractualTerms' hasExplicitDutyCounterpart='Consumer'
art5;par2	Procedure	hasProcedureBearer='Supplier' hasProcedureAction='Communication' hasProcedureObject='ContractualTerms' hasProcedureCounterpart='Consumer'
art5;par3	ExplicitRight	hasExplicitRightBearer='Consumer' hasExplicitRightCounterpart='Supplier'
art6;par1	ExplicitDuty	hasExplicitDutyBearer='EUMemberState' hasExplicitDutyCounterpart='Consumer'
art7;par2	ExplicitPower	hasExplicitPowerBearer='EUMemberState' hasExplicitPowerCounterpart='Supplier'
art11;par1	ExplicitPower	hasExplicitPowerBearer='EUMemberState' hasExplicitPowerCounterpart='Supplier'

Table 1

Semantics of Directive 2002/65/EC excerpt in functional notation

Having defined the `prv` and `cl` namespaces[13]

```
xmlns:prv="http://www.ittig.cnr.it/ontologies/def/ProvisionModel#"
xmlns:cl="http://www.ittig.cnr.it/ontologies/def/ConsumerLaw#"
```

[11] http://godel.ittig.cnr.it/docs/Directive2002-65-EC-Excerpt.xml

[12] here the URN-LEX convention for fragments identification is used (http://datatracker.ietf.org/docs/draft-spinosa-urn-lex/)

[13] according to the recommendations given by the European Commission in "Study on persistent URIs, with identification of best practices and recommendations on the topic for the MSs and the EC", within the ISA Programme

for Provision Model and DALOS consumer law domain ontology, respectively, an RDF/XML[14] semantic annotation of art. 5 paragraph 1, reported as example of Directive annotation[15], can be the following:

```
<rdf:Description rdf:about="[URI]#art5;par1">
 <rdf:type rdf:resource="prv:ExplicitDuty"/>
 <prv:hasExplicitDutyBearer rdf:resource="cl:Supplier"/>
 <prv:hasExplicitDutyAction rdf:resource="cl:Communication"/>
 <prv:hasExplicitDutyObject rdf:resource="cl:ContractualTerms"/>
 <prv:hasExplicitDutyCounterpart rdf:resource="cl:Consumer"/>
</rdf:Description>
```

5.2. *Querying the system*

Having an OWL-DL description of the Provision Model and provision instances, a provisions management system can be given inference facilities through an OWL-DL reasoner able to derive an inferred model. In this example the Pellet[16] Java based OWL-DL reasoner is used. The result is a Provision Model where inferences are calculated from the associated axioms. At this stage an RDF triple store of provisions can be queried using SPARQL[17]. Let's assume to query the Directive excerpt in Section 5.1 in order to demonstrate the approach and, as first example, a query able to retrieve consumer's rights:

```
SELECT ?x WHERE  { ?x  prv:hasRightBearer cl:Consumer }
```

where `?x` is the variable that will contain the identifiers of the retrieved provisions instances (usually paragraphs).

In case the *non-inferred model* is queried, no provisions are retrieved since only ExplicitRight and related attributes are used for provision annotation. To obtain the rights explicitly expressed, the query has to be specified asking for provisions whose hasExplicitRightBearer value is cl:Consumer. In this case, paragraph with `id="art5;par3"` is correctly retrieved.

In case the *inferred model* is queried, all the inferred provisions are retrieved, either annotated as ExplicitRight of Consumer or implicitly deduced by provision relations. Since Hohfeldian relations have been implemented in the Provision Model, the result will be an Hohfeldian reasoning over provisions. By exploiting the established rdfs:subClass and owl:equivalentClass relations between provisions type and attributes, the system will act as virtually expanding the query and obtaining the results as shown in Tab. 2.

Virtual SPARQL query expansion		Result
`?x`	`prv:hasExplicitRightBearer` `cl:Consumer`	art5;par3
`?x`	`prv:hasImplicitRightBearer` *[≡ prv:hasExplicitDutyCounterpart]* `cl:Consumer`	art5;par1 art6;par1

Table 2

Virtual SPARQL query expansion corresponding to inferences on provision axioms. The *[≡ prv:hasExplicitDutyCounterpart]* expression represents the equivalence relation prv:hasImplicitRightBearer ≡ prv:hasExplicitDutyCounterpart.

Moreover, the distinction between implicit/explicit provisions and attributes allows to select, for example, among the *Rights* of a *Bearer*, only those not explicitly expressed in the text. The related query will be:

```
SELECT ?x WHERE {?x prv:hasImplicitRightBearer cl:Consumer}
```

which will retrieve the ExplicitDuty individuals where hasExplicitCounterpart is Consumer (being hasImplicitRightBearer ≡ hasExplicitDutyCounterpart); in the example of Section 5.1 the following paragraphs are retrieved: `id="art5;par1", id="art6;par1"`. Similar considerations can be made about querying for potestative provisions, like consumer's powers.

6. Technical relations between provisions

As previously introduced (Section 3) another kind of relations between provisions can be identified: we call them *technical relations* because they are not necessary from a logical point of view, but they are possible and derive from legislative techniques considerations. Such relations can be detected at the level of provision instances only.

An example of them can be the relation existing between a *Duty* of a *Bearer* to accomplish a specific *Action* towards a *Counterpart*, the *Procedure* describing how to fulfill such duty, the *Exceptions* to it, as well as the *Sanction* such *Bearer* may face if he does not fulfill such duty. In the excerpt of Directive 2002/65/EC in Section 5.1, Art. 5 paragraph 1 and 2 represent a *Duty* and the correlated *Procedure* (see also Tab. 1).

Another example of technical relations between provisions can be the one existing between a *Term Definition* introducing a specific entity through its *Definien-*

[14]XML serialization of the Resource Description Framework language for semantic annotation of the resources
[15]http://godel.ittig.cnr.it/docs/Directive2002-65-EC-Excerpt.rdf
[16]http://clarkparsia.com
[17]query language for RDF

dum and other provisions involving the same entity. In the excerpt of Directive 2002/65/EC in Section 5.1, Art. 2 paragraph 1 letter c) and Art. 5 paragraph 1 and 2, as well as other provisions involving the 'Supplier', represent correlated provision instances.

In terms of Provision Model such relations can be established by the legislator through references or deduced by reasoning over provisions content and detected through the identity between a number of values in corresponding attributes of different provisions. In this respect [7] distinguishes between *strong* and *weak* relations between provision instances according to whether there is identity between all the values in corresponding attributes (*strong* relations) or only between some of them (*weak* relations). The number of attributes and values in identity relation gives the degree of strength/weakness of the relation itself [7].

Technical relations between provision instances are particularly interesting for providing users with advanced retrieval services. For example, while querying for his duties, a supplier might be also informed about the procedures to fulfill such a duty and possible sanctions in case of non-compliance.

7. Provision technical relations example

In this section a possible reasoning implementation dealing with technical relations between provisions is shown. On the basis of the semantic annotation proposed in Section 5.1, an example of a SPARQL query able to retrieve the supplier's duties is:

```
SELECT ?x WHERE  { ?x  prv:hasDutyBearer cl:Supplier }
```

Firstly, in case the *inferred model* is queried, the paragraphs with `id="art5;par1"` and `id="art5;par3"` are retrieved, representing both explicit and implicit duties of the 'Supplier', thus implementing a Hohfeldian reasoning over provisions. At this stage the system can analyse the attribute values of the retrieved provisions and construct a query able to check whether correlated provisions of ExplicitDuty at `id="art5;par1"` of type Procedure are available (the same can be done for the ImplicitDuty at `id="art5;par3"`). Such relation involves all the attribute values of the correlated provisions (*strong* relation), therefore the query able to retrieve such correlated provisions can be the following:

```
SELECT ?x WHERE
{
   {?x prv:hasDutyBearer cl:Supplier.
    ?x prv:hasDutyAction cl:Communication.
    ?x prv:hasDutyObject cl:ContractualTerms.
    ?x prv:hasDutyCounterpart cl:Consumer
   }
   UNION
   {?x prv:hasProcedureBearer cl:Supplier.
    ?x prv:hasProcedureAction cl:Communication.
    ?x prv:hasProcedureObject cl:ContractualTerms.
    ?x prv:hasProcedureCounterpart cl:Consumer
   }
}
```

This query will retrieve the Duty provision instance with `id="art5;par1"` (in this case ExplicitDuty) and the correlated Procedure with `id="art5;par2"`. Such query can also be extended to search for correlated sanctions (provision type Redress).

8. Conclusions

The combination of Provision Model and domain ontologies can represent an approach for semantic annotation of legislative texts, with the aim of providing advanced retrieval and reasoning facilities over norms by exploiting relations between provisions. In this paper an approach has been proposed to describe logical relations between provisions, as the Hohfeldian fundamental relations, using OWL-DL: it is implemented by extending the Provision Model to represent either implicit or explicit provision types and attributes. Similarly, an approach dealing with technical relations between provisions, involving provision types, attributes and attribute values, has been presented.

At this stage of development this work represents a contribution to identify reasoning schemas that can be dealt within a DL computational tractability by OWL-DL, thus exploiting existing DL reasoners, without using SWRL or RIF as in [19] or rules description using specific XML schemas, as in [16]. The identification of the sufficient conditions within which legal reasoning can be kept within a DL complexity represents a possible future development of this work.

On the other hand a drawback of the approach is the need to rely on legislative texts properly marked-up into fragments or formal partitions (structural mark-up), as well to qualify them in terms of provisions (semantic mark-up). The availability of XML mark-up of legislative texts into formal partitions is being implemented at institutional levels within several legislative XML projects [13] [27] and more recently within Linked Open Data initiatives. In order to guarantee the scalability of the approach, the use of software tools supporting the activity of structural mark-up is highly recommended: they can be word processors (editors) able to support mark-up activities [2] [29] [22], or textual parsers able to detect legislative documents structure [4]. The qualification of such resources in terms of provisions semantics (semantic mark-up) is a more complex task, that can represent a burden for legislators or for documentalists, because it is an intellectual activity which is subject to different interpretation especially as regards the identification of the actual meaning of the norms. Therefore, it is not a task that can be accomplished by an (semi-)automatic transformation of a formal XML markup into RDF[18], but entails an additional level of interpretation which still needs a human intervention. As for the structural markup, also such provisions semantic mark-up can be supported by software tools, usually based on machine learning and NLP techniques, able to automatically (or semi-automatically) classify provisions [15] [12] and extract the related attributes [8] [14].

The approach presented in this paper can be effectively used to provide a semantic annotation refinement to legislative documents, published according to the Linked Open Data principles, thus delegating to different actors, as public administrations, the burden of providing legislative documents at a minimum level of interoperability, including XML structural mark-up.

References

[1] T. Agnoloni, L. Bacci, E. Francesconi, W. Peters, S. Montemagni, and G. Venturi, *A two-level knowledge approach to support multilingual legislative drafting*, in J. Breuker, P. Casanovas, M. Klein, and E. Francesconi, editors, Law, Ontologies and the Semantic Web, volume 188 of Frontiers in Artificial Intelligence and Applications, pages 177–198. IOS Press, Amsterdam, The Netherlands, 2009.

[2] T. Agnoloni, E. Francesconi, and P. Spinosa, *xmLegesEditor: an opensource visual xml editor for supporting legal national standards*, in Proceedings of the V Legislative XML Workshop, pages 239–251. European Press Academic Publishing, Florence, Italy, 2007.

[3] G. Antoniou, D. Billington, G. Governatori, and M. Maher. *On the modeling and analysis of regulations*, in Proceedings of the Australian Conference Information Systems, Victoria University of Wellington, New Zealand, pages 20–29, 1999.

[4] L. Bacci, P. Spinosa, C. Marchetti, and R. Battistoni. *Automatic mark-up of legislative documents and its application to parallel text generation*, in N. Casellas, E. Francesconi, R. Hoekstra, and S. Montemagni, editors, *Proceedings of the 3rd Workshop on Legal Ontologies and Artificial Intelligence Techniques joint with 2nd Workshop on Semantic Processing of Legal Texts*, pages 45–54, Huygens Editorial, Barcelona, Spain 2009.

[5] J. Bentham. H. L. A. Hart, ed. *Of Laws in General*. University of London, Athlone Press, 1970, ISBN 0485132109.

[6] C. Biagioli. *Law making environment: model based system for the formulation, research and diagnosis of legislation*, 1996, http://www.ittig.cnr.it/Ricerca/Testi/Biagioli96AI-Law.pdf (Retrieved on 2 September 2014).

[7] C. Biagioli. *Modelli Funzionali delle Leggi. Verso testi legislativi autoesplicativi.*, volume 6 of Legal Information and Communications Technologies Series, European Press Academic Publishing, Florence Italy, 2009, ISBN 9788883980596

[8] C. Biagioli, E. Francesconi, A. Passerini, S. Montemagni, and C. Soria. *Automatic semantics extraction in law documents*, in International Conference on Artificial Intelligence and Law, pages 133–139, Bologna, Italy. ACM 2005 ISBN 1-59593-081-7.

[9] C. Biagioli and D. Grossi. *Formal aspects of legislative meta-drafting*, in Proceedings of the Jurix Conference: Legal Knowledge and Information Systems, pages 192–201, IOS Press, The Netherlands, 2008.

[10] A. Boer, R. Hoekstra, E. de Maat, E. Hupkes, F. Vitali, and M. Palmirani. *CEN Workshop Agreement "Metalex (Open XML Interchange Format for Legal and Legislative Resources)'*, n. CWA 15710:2010 E, CEN European Committee for Standardization, 2010.

[11] A. G. Conte. *Filosofia del linguaggio normativo. II. Studi 1982-1994*, Giappichelli, Torino, 1995.

[12] E. de Maat, K. Krabben, and R. Winkels. *Machine learning versus knowledge based classification of legal texts*, in Proceedings of the Jurix Conference: Legal Knowledge and Information Systems, pp. 87-96, IOS Press, The Netherlands, 2010.

[13] E. Francesconi. *Technologies for European Integration. Standards-based Interoperability of Legal Information Systems*, in Legal Information and Communication Technologies Series, vol. European Press Academic Publishing, Florence, Italy, 2007, ISBN 978-88-8398-050-3.

[14] E. Francesconi. *An approach to legal rules modelling and automatic learning*, In G. Governatori, editor, Proceedings of the International Conference on Legal Knowledge and Information Systems, volume 205 of Frontiers in Artificial Intelligence and Applications, pages 59–68. IOS Press, 2009.

[15] E. Francesconi and A. Passerini. *Automatic classification of provisions in legislative texts*, International Journal on Artificial Intelligence and Law, 15(1):1-17, Springer, Berlin, 2007.

[16] T. Gordon. *Combining rules and ontologies with Carneades*, in Proceedings of the 5th International RuleML2011@BRF Challenge, volume 799. CEUR-WS.org, 2011.

[17] H. Hart. *The Concept of Law*, Clarendon Law Series. Oxford University Press, 1961.

[18] P. Hitzler, M. Krötzsch, and S. Rudolph. *Foundations of the Semantic Web Technologies*, Chapman & Hall/CRC, 2009.

[19] R. Hoekstra, J. Breuker, M. di Bello, and A. Boer. *LKIF core: Principled ontology development for the legal domain*, In J. Breuker, P. Casanovas, M. Klein, and E. Francesconi, editors, Law, Ontologies and the Semantic Web, pp. 21-52, vol. 188 – Frontiers in Artificial Intelligence and Applications, IOS Press, The Netherlands, 2009.

[20] W. N. Hohfeld. *Some fundamental legal conceptions as applied in judicial reasoning. I*, in Yale Law Journal, 23:16–59, 1913.

[21] H. Kelsen. *General Theory of Norms*, Clarendon Press, Oxford, 1991.

[22] M. Palmirani and F. Benigni. *Norma-system: A legal information system for managing time*, in Proceedings of the V Legislative XML Workshop, pages 205–223. European Press Academic

[18] For example http://rhizomik.net/html/redefer/xml2rdf/

Publishing, Florence, Italy, 2007.

[23] J. Rawls. *Two concepts of rules*, in Philosophical Review, 64:3–32, 1955.

[24] J. Raz. *The Concept of a Legal System*, Oxford University Press, 1980.

[25] A. Ross, *Directives and Norms*, Routledge & K. Paul, London, 1968.

[26] G. Sartor. *Fundamental legal concepts: A formal and teleological characterisation*, in Artificial Intelligence and Law, 14(1-2):101–142, 2006.

[27] G. Sartor, M. Palmirani, E. Francesconi, and M. Biasiotti, editors. *Legislative XML in the Semantic Web. Principles, Models, Standards for Document Management*, ISBN 978-94-007-1886-9 in Law, Governance and Technology Series. Springer, Berlin, 2011.

[28] J. Searle. *Speech Acts: An Essay in the Philosophy of Language*, Cambridge University Press, 1969, ISBN 978-0521096263

[29] S. van de Ven, R. Hoekstra, R. Winkels, E. de Maat, and A. Kollar. *MetaVex: Regulation drafting meets the semantic web*, in Computable Models of the Law. Languages, Dialogues, Games, Ontologies, pp. 42-55, Lecture Notes in Computer Science, 4884, Heidelberg-Berlin: Springer Verlag, Berlin, 2007