A Deontic Logic Semantics for Licenses Composition in the Web of Data

Antonino Rotolo
University of Bologna
Italy
antonino.rotolo@unibo.it

Serena Villata*
INRIA Sophia Antipolis
France
serena.villata@inria.fr

Fabien Gandon
INRIA Sophia Antipolis
France
fabien.gandon@inria.fr

ABSTRACT

In the Web of Data, the absence of clarity about the licensing terms under which the data is released prevents data reuse, and thus data publication and interlinking at the expenses of the Web of Data itself. In addition, even when terms are clear, the absence of automated processing of the licenses prevents scaling data reuse and integration. In this paper, we provide a semantic model of licenses for the Web of Data. The key idea of our approach consists first in verifying the compatibility and compliance of the licensing terms associated to the data queried by the consumer, and second, if compatibility arises, in composing the single licenses into a unique license which provides the terms of reuse for the whole data consumed during the query solving. In particular, we propose a deontic logic semantics which is able to (i) formally define the deontic components of the licenses, i.e., *Permissions*, *Obligations*, and *Prohibitions*, and reason over them, (ii) verify the compatibility of the elements composing the single licenses, and return those elements which can be included into the composite license, and (iii) provide a formal account of the heuristics proposed to guide the composition.

Categories and Subject Descriptors

I.2.4 [**Artificial Intelligence**]: Knowledge Representation Formalisms and Methods

General Terms

Theory

Keywords

Data licensing, Semantic Web

*The author acknowledges support of the DataLift Project ANR-10-CORD-09 founded by the French National Research Agency.

1. INTRODUCTION

The Web is evolving from an information space for sharing textual documents into a medium for publishing *structured* data too. The Linked Data[1] initiative aims at fostering the publication and interlinking of data on the Web, giving birth to the so called *Web of Data*, an interconnected global dataspace where data providers publish their content publicly or under open licenses. Heath and Bizer [17] underline that "the absence of clarity for data consumers about the terms under which they can reuse a particular dataset, and the absence of common guidelines for data licensing, are likely to hinder use and reuse of data". Therefore, all Linked Data on the Web should include explicit licensing terms, or waiver statements [21]. The explicit definition of the licensing terms under which the data is released is an open problem both for data providers and for data consumers [2]. The former needs to explicit the licensing terms to ensure use and reuse of the data compliant with her requirements. The latter, instead, needs to know the licenses constraining the released data to avoid misusing and even illegal reuse of such data.

Our objective is to start from the different licenses collected along with the query result, and build a so called *composite license* which combines the elements from the distinct licenses if they are evaluated to be compliant with each other. We start from the recent proposal of Villata and Gandon [24] which considers the Creative Commons (CC) licenses schema [1] only, and we go beyond it by capturing, formally specifying and reasoning over the major deontic components of the licenses in the Web of Data. The research question we answer in this paper is: *how to specify and reason over the deontic components of the licenses in the Web of Data?* This question breaks down into the following subquestions: (1) How to verify the compatibility among the elements composing the single licenses associated to a query result?, and (2) How to specify and evaluate the heuristics to compose, if compatible, these distinct elements into a composite license? We answer the research questions by adopting standard Semantic Web languages like RDF and OWL for representing the concepts and attaching the composite license to the query result, and defeasible deontic logic to verify the compatibility, and build the composite license following specific heuristics.

First, we verify the compatibility of the elements composing the single licenses using our deontic logic semantics. The proposed defeasible logic verifies the presence of possible conflicts between the rules of the different licenses, and it returns the set of obligations, permissions and prohibitions

[1]`http://linkeddata.org`

that belong to the composite license l_c. If conflicts cannot be solved, and no priorities are defined among the licenses, then the resulting composite license is empty, i.e., it does not imply any obligation, prohibition and permission.

Second, if the licenses are compatible, the logic returns the composite license. Three main heuristics [8, 24] are considered and formally defined to compose the licenses into a single one, namely *OR-composition*, *AND-composition*, and *Constraining-value*. Data providers may use our deontic logic semantics to simulate the behavior of the heuristics to verify which one better satisfies their intended behavior. The machine-readable version of the composite license is then generated using the required licenses schema, and it is returned together with the query result.

In this paper, we consider Web of Data (possibly machine-readable) license specifications only, and we do not consider the MPEG-21 Rights Expression Language[2]. Moreover, we do not provide any kind of verification about the goodness of the reuse of the data performed by the consumer. This is a debated open issue, we will address it as future research.

Our approach is to (i) use a combination of Semantic Web languages and defeasible deontic logic, (ii) extend and adapt existing proposals for licenses compatibility and composition in the area of service license analysis [10] and Creative Commons licenses [24] to the Web of Data scenario, and (iii) verify which heuristics better suits data provider's needs and allow their eventual combination, if desired.

The remainder of the paper is as follows: Section 2 presents the problem of data licensing in the Web of Data as well as the proposed framework for licenses composition. Section 3 defines our deontic logic semantics and Section 4 compares the proposed approach with existing research.

2. LICENSES IN THE WEB OF DATA

The Web is evolving from an interconnection of HTML pages to the *Web of Data*, a global dataspace of content currently locked in relational databases. The Semantic Web community aims at "enabling computers to do more useful work and to develop systems that can support trusted interactions over the network"[3]. Despite small differences, names such as Semantic Web, Web of Data and Linked Data all refer to world-wide initiatives focused on nurturing and managing interconnected datastores on the Web. The Linked Data initiative is effectively contributing to the growth of the Web of Data, favoring the publication and the interlinking of billions of RDF triples from heterogeneous domains (media, user-generated content, government, life science, etc).

The design issues of Linked Data[4] highlight, as of the first star, the need not only to publish the data on the Web, but to publish it with an *open license* in order to be Open Data. As a consequence, several (possibly machine-readable) licensing terms for the data have been proposed, leading to a huge amount of unrelated but comparable ways of expressing the licenses. The scenario we consider in this paper is the following: when data consumers query the Web of Data, results from different datasets, and thus released under different licensing terms, are provided; the objective of this paper is to study how to build a composite license which combines the compatible elements from the distinct licenses associated to the data selected by the consumer's query.

We start from the approach proposed by Villata and Gandon [24] for licenses composition in the Web of Data: the basic idea is first to verify through a number of rules whether the elements composing the single licenses are *compatible*, and if a positive answer follows, these elements are *composed* into a composite license which is the unique license returned together with the queried data. A main feature characterizing the work of Villata and Gandon [24] is the use of Semantic Web languages, i.e., RDF and OWL, to allow the development of a data licensing module as envisioned by the W3C Provenance Working Group[5]. However, this work presents the following drawbacks: (i) the proposed model considers uniquely CC licenses without taking into account other licenses like Open Data Commons (ODC) and waivers, (ii) the compatibility among the elements composing the licenses is provided using a set of *compatibility rules* which are *strict* and do not consider the *counts-as* relation among the elements, and (iii) the heuristics for licenses composition are only informally presented. In this paper, we aim at addressing these open issues by providing a deontic logic semantics which allows us (i) to define a general framework where CC licenses are just one possible instance, (ii) to capture and study the deontic components of our licenses, and (iii) to formally define the heuristics guiding licenses composition.

Our semantic model for licenses in the Web of Data relies on the following basic licenses' structure [24]. Each license l_i is composed by a set of models that are *Obligations, Permissions* and *Prohibitions*. The models are finally composed by elements which specify the precise obligation, permission or prohibition which is included in the license, e.g., the *Attribution* and *Share-Alike* obligations are the mostly diffused ones in the licenses considered by Open Knowledge Definition[6].

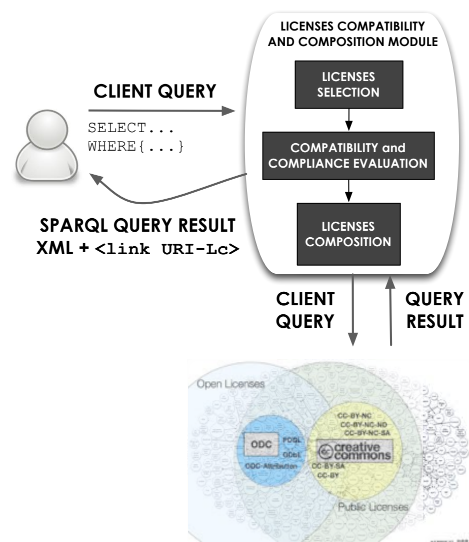

Figure 1: The licenses composition framework.

The proposed framework for licenses composition works as follows (Figure 1): (i) the data consumer queries the Web of Data where the datasets are connected to different licenses[7],

[2]http://bit.ly/MPEG-21
[3]http://www.w3.org/standards/semanticweb/
[4]http://www.w3.org/DesignIssues/LinkedData.html
[5]http://www.w3.org/2011/prov/wiki/Main_Page
[6]http://opendefinition.org/licenses/
[7]We follow the proposal of [24] by choosing named graphs as

e.g., ODC, CC, GNU Free Documentation License; (ii) the results of the query (possibly) return data from different data sources and released under different licensing terms; (iii) our module first collects all the URIs of the licenses attached to the data selected by the query, second it verifies, using our deontic logic semantics, whether there are conflicts among these licenses, and third, if the licenses are compatible, the framework generates the composite license using one of the three heuristics, or a composition of them; (iv) the composite license is generated in RDF, and it is associated to a URI, and (v) the URI of the composite license is returned to the consumer together with the query results using the standard SPARQL query results XML format.

Given the need to express the licenses in a machine-readable format, we propose a lightweight vocabulary called `l4lod`[8] (Licenses for Linked Open Data) for expressing the licensing terms in the Web of Data, visualized in Figure 2. In particular, we define the class `License` which is a sub-class of `dc:LicenseDocument`[9] (as for the class `cc:License`), and three basic deontic properties which are respectively **`permits`**, **`prohibits`**, and **`obliges`**. These properties connect each license with its own elements: `Reproduction, Derivative, Distribution, Sharing` (for *Permissions*), `Attribution, ShareAlike` (for *Obligations*), and `Commercial` (for *Prohibitions*). The `l4lod` vocabulary is not intended to propose yet another license, but it is intended to provide the basic means to define in a machine-readable format the existing licensing terms such as those expressed by ODC licenses, the GNU Free Documentation License, etc. The vocabulary does not provide an exhaustive set of properties for licenses definition. Implementations are free to extend `l4lod` to add further elements. The `licensingTerms` property is introduced to connect the machine-readable description of the license to its human-readable counterpart. We go beyond the proposal of [24] who choose the CC vocabulary as a general schema for every kind of license specification. This is motivated by the observation that there are works which cannot be considered as *creative work*, and thus should not be released under CC licenses [17]. Moreover, the vocabulary considers the alignment with the following vocabularies: the CC vocabulary[10], the Dublin Core vocabulary[9], the Waiver vocabulary[11], the Description of a Project vocabulary (`doap`)[12], the Ontology Metadata vocabulary (`omv`)[13], the Data Dictionary for Preservation Metadata (`premis`)[14], the Vocabulary Of Attribution and Governance (`voag`)[15], the NEPOMUK Information Element ontology (`nie`)[16], the Music Ontology (`mo`)[17], the Good Relations vocabulary (`gr`)[18], and myExperiment Base ontology (`meb`)[19].

The example in Figure 3 shows a well-known license expressed using RDF. In particular, the license `lic1` expresses the ODC Attribution license[20] which *permits* to copy, distribute and use the database (i.e. sharing) and obliges for Attribution.

granularity level to attach the licensing terms to the data.

[8] http://ns.inria.fr/l4lod/

[9] http://purl.org/dc/terms/

[10] http://creativecommons.org/ns

[11] http://vocab.org/waiver/terms/.html

[12] http://usefulinc.com/ns/doap

[13] http://omv2.sourceforge.net/index.html

[14] http://bit.ly/premisOntology

[15] http://voag.linkedmodel.org/schema/voag

[16] http://bit.ly/nieOntology

[17] http://purl.org/ontology/mo/

[18] http://purl.org/goodrelations/v1

[19] http://rdf.myexperiment.org/ontologies/base/

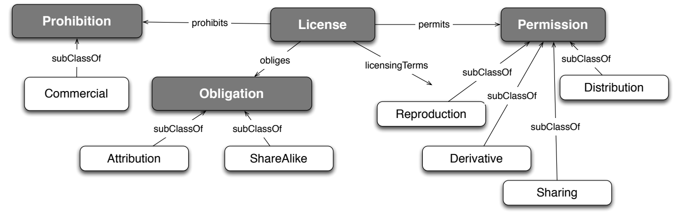

Figure 2: The `l4lod` lightweight vocabulary.

```
@prefix l4lod: http://ns.inria.fr/l4lod/.
@prefix : http://example/licenses.

:lic1 a l4lod:License;
    l4lod:licensingTerms
    <http://opendatacommons.org/licenses/by/>;
    l4lod:permits l4lod:Sharing;
    l4lod:obliges l4lod:Attribution.
```

Figure 3: The license `lic1`.

3. THE LOGIC

The following is an extension of DL, which revises earlier works [15, 16]. Like in [14, 3, 4], the formal language is designed to represent, and reason about two components: a first one is meant to describe the ontology of concepts involved in Web of Data licenses; the second one aims to capture the deontic component of those licenses.

The formal language is rule-based. Literals can be plain, such as $p, q, r \ldots$, or modal, such O (obligatory), P (permitted), and F (forbidden/prohibited).

Ontology rules work as regular DL rules for deriving plain literals, while the logic of deontic rules provide a constructive account of the basic deontic modalities (obligation, prohibition, and permission). Notice, however, that the purpose of the formalism is to establish the conditions to derive ontology and deontic conclusions from different licenses, so we need to keep track of how these conclusions are obtained. To this purpose, rules (and, as we will see, their conclusions) are parametrized by labels referring to licenses.

An ontology rule such as $a_1, \ldots, a_n \Rightarrow_c^{l1} b$[21] support the conclusion of b, given $a_1, \ldots, a_n$, and so states that, from the viewpoint of license l_1 any instance enjoying $a_1, \ldots, a_n$ is also an instance of b; rules such as $a, \mathrm{O}b \Rightarrow_{\mathrm{O}}^{l2} p$ states that, if a is the case and b is obligatory, then $\mathrm{O}p$ holds in the perspective of license l_2, i.e., p is obligatory for l_2.

The proof theory we propose aims at combining licenses, checking their compatibility, and establishing what ontology and deontic conclusions can be drawn from the composite license. In other words, if $l_c = l_1 \odot \cdots \odot l_n$ is the composite license obtained from $l_1, \ldots, l_n$, the conclusions derived in the logic are those that hold in the perspective of l_c.

[20] http://opendatacommons.org/licenses/by/

[21] The subscript c stands for "concept rule". Elsewhere, we have called this type of rules constitutive or counts-as rules [16, 14, 3, 4].

3.1 Formal Language and Basic Concepts

The basic language is defined as follows. Let $\mathrm{Lic} = \{l_1, l_2, \ldots, l_n\}$ be a finite set of licenses. Given a set PROP of *propositional atoms*, the set of *literals* Lit is the set of such atoms and their negation; as a convention, if q is a literal, $\sim q$ denotes the complementary literal (if q is a positive literal p then $\sim q$ is $\neg p$; and if q is $\neg p$, then $\sim q$ is p). Let us denote with $\mathrm{MOD} = \{\mathrm{O}, \mathrm{P}, \mathrm{F}\}$ the set of basic deontic modalities. The set ModLit of modal literals is defined as follows: i) if $X \in \mathrm{MOD}$ and $l \in \mathrm{Lit}$ then Xl and $\neg Xl$ are modal literals, ii) nothing else is a modal literal.

Let Lbl be a set of arbitrary labels. Every rule is of the type $r : A(r) \hookrightarrow^x_Y C(r)$, where

1. $r \in \mathrm{Lbl}$ is the name of the rule;
2. $A(r) = \{a_1, \ldots, a_n\}$, the *antecedent* (or *body*) of the rule, is a finite set denoting the premises of the rule. If r is an ontology rule, then each a_i, $1 \leq i \leq n$, belongs to Lit, otherwise it belongs to $\mathrm{Lit} \cup \mathrm{ModLit}$;
3. $\hookrightarrow \in \{\rightarrow, \Rightarrow, \rightsquigarrow\}$ denotes the type of the rule;
4. $Y \in \{c, \mathrm{O}, \mathrm{P}\}$ represents the type of conclusion obtained[22];
5. if $Y = \mathrm{P}$, then $\hookrightarrow \in \{\rightarrow, \Rightarrow\}$;
6. $x \in \mathrm{Lic}$ indicates to which license the rule refers to;
7. $C(r) = b \in \mathrm{Lit}$ is the *consequent* (or *head*) of the rule.

The intuition behind the different arrows is the following. *Strict rules* have the form $a_1, \ldots, a_n \rightarrow^x_Y b$. *Defeasible rules* have the form $a_1, \ldots, a_n \Rightarrow^x_Y b$. A rule of the form $a_1, \ldots, a_n \rightsquigarrow^x_Y b$ is a *defeater*. The three types of rules establish the strength of the relationship. Strict rules provide the strongest connection between a set of premises and their conclusion: whenever the premises are deemed as indisputable so is the conclusion. Defeasible rules allows to derive the conclusion unless there is evidence for its contrary. Finally, defeaters suggest that there is a connection between its premises and the conclusion not strong enough to warrant the conclusion on its own, but such that it can be used to defeat rules for the opposite conclusion[23].

A multi-license theory is the knowledge base which is used to reason about the applicability of license rules under consideration.

Definition 1. *A* multi-license theory *is a structure* $D = (F, L, \{R^l\}_{l\in\mathrm{Lic}}, \{R^{\mathrm{O}^l}\}_{l\in\mathrm{Lic}}, \{R^{\mathrm{P}^l}\}_{l\in\mathrm{Lic}}, \succ)$, *where*

- $F \subseteq \mathrm{Lit} \cup \mathrm{ModLit}$ *is a finite set of facts;*
- $L \subseteq \mathrm{Lic}$ *is a finite set of licenses;*
- $\{R^l\}_{l\in\mathrm{Lic}}$, $\{R^{\mathrm{O}^l}\}_{l\in\mathrm{Lic}}$, *and* $\{R^{\mathrm{P}^l}\}_{l\in\mathrm{Lic}}$ *are finite families of sets of ontology and obligation rules, respectively;*
- $\succ$ *is an acyclic relation (called superiority relation) defined over* $(R^l \times R^{l'}) \cup ((R^{\mathrm{O}^l} \cup R^{\mathrm{P}^l}) \times (R^{\mathrm{O}^{l'}} \cup R^{\mathrm{P}^l}))$, *where* $R^l, R^{l'} \in \{R^l\}_{l\in\mathrm{Lic}}$, $R^{\mathrm{O}^l}, R^{\mathrm{O}^{l'}} \in \{R^{\mathrm{O}^l}\}_{l\in\mathrm{Lic}}$, $R^{\mathrm{P}^l}, R^{\mathrm{P}^{l'}} \in \{R^{\mathrm{P}^l}\}_{l\in\mathrm{Lic}}$ *and* $l \neq l'$[24].

$R[b]$ and $R^X[b]$ with $X \in \{l, \mathrm{O}^l, \mathrm{P}^l | l \in \mathrm{Lic}\}$ denote the set of all rules whose consequent is b and of all rules (of type X). The sets R_s, R_{sd}, and R_{dft} denote in D, respectively, the sets of strict rules, defeasible rules, and defeaters.

3.2 Proof Theory

A *proof* P of length n is a finite sequence $P(1), \ldots, P(n)$ of tagged literals of the type $+\Delta^X q$, $-\Delta^X q$, $+\partial^X q$ and $-\partial^X q$, where $X \in \{l, Y^l | l \in \mathrm{Lic}, Y \in \mathrm{MOD}\}$. The proof conditions below define the logical meaning of such tagged literals. As a conventional notation, $P(1..i)$ denotes the initial part of the sequence P of length i. Given a multi-license theory D, $+\Delta^X q$ means that literal q is provable in D with the mode X using only facts and strict rules, $-\Delta^X q$ that it has been proved in D that q is not definitely provable in D with the mode X, $+\partial^X q$ that q is defeasibly provable in D with the mode X, and $-\partial^X q$ that it has been proved in D that q is not defeasibly provable in D with the mode X.

Given $\sharp \in \{\Delta, \partial\}$, $P = P(1), \ldots, P(n)$ *is a proof for* p in D for the license l iff $P(n) = +\sharp^l p$ when $p \in \mathrm{Lit}$, $P(n) = +\sharp^{X^l} q$ when $p = Xq \in \mathrm{ModLit}$, and $P(n) = -\sharp^{Y^l} q$ when $p = \neg Yq \in \mathrm{ModLit}$.

The proof conditions below aim at determining what conclusions can be obtained within composite licenses by using the source licenses. As we have recalled in Section 2, three heuristics have been proposed for this purpose [8, 24]:

- **OR-composition:** if there is at least one of the licenses involved in the composition that owns a clause then also l_c owns it;
- **AND-composition:** if all the licenses involved in the composition own a clause then also l_c owns it;
- **Constraining-value:** the most constraining clause among those offered by the licenses is included in l_c.

While the first two heuristics can be directly implemented within the proof theory, the third one requires an additional notion. We will outline a simple way for implementing it at the end of this section. Notice that OR-composition and AND-composition were proposed in [24] to only model the deontic part of licenses, i.e., the one meant to derive obligations, permissions, and conclusions. Here, we will show that it makes sense to apply it also to the ontology part. The constraining-value heuristics instead applies only to the deontic effects of licenses.

Some notational conventions and concepts that we will use throughout the remainder of this section:

- Let $l_c = l_1 \odot \cdots \odot l_n$ be any composite license that can be obtained from the set of licenses $L_c = \{l_1, \ldots, l_n\} \subseteq L$;

[22] We will see why we do not need rules for prohibitions.

[23] This is the reason why we do not have defeaters for permission. For a discussion, see [13].

[24] Our purpose is to develop a reasoning method for specifically handling license composition. This is reflected in the fact that $l \neq l'$: in other words, we assume to work on, and solve only conflicts between rules for different licenses. Relaxing this restriction is harmless and allows us to also solve conflicts within licenses.

- Let $X, Y \in \mathrm{MOD}$.

DEFINITION 2. *For any pair of rule sets R and S, we define as follows the operation $\sqcap$ on them*

$$R \sqcap S = \{r : a_1, \ldots, a_n \hookrightarrow^x_Y b,\ s : a'_1, \ldots, a'_n \rhd^j_P b' | \\ r \in R, s \in S, A(r) = A(s), \hookrightarrow = \rhd, P = Y, \\ C(r) = C(s)\}$$

As usual with DL, we have proof conditions for the monotonic part of the theory (proofs for the tagged literals $\pm\Delta^Y p$) and for the non-monotonic part (proofs for the tagged literals $\pm\partial^Y p$). To check licenses' compatibility and compose them means to apply the proof conditions of the logic to a multi-license where the set of licenses is $L = L_c$.

3.2.1 Definite Provability

The definitions below for Δ describe just forward (monotonic) chaining of strict rules.

Ontology Definite Provability.

$+\Delta^{l_c}$: If $P(n+1) = +\Delta^{l_c} q$ then
(1) $q \in F$ or
(2) $\exists r \in R^x_s[q] :\ \forall a \in A(r)$,
$+\Delta^{l_c} a \in P(1..n)$.

$-\Delta^{l_c}$: If $P(n+1) = -\Delta^{l_c} q$ then
(1) $q \notin F$ and
(2) $\forall r \in R^x_s[q] :\ \exists a \in A(r)$,
$-\Delta^{l_c} a \in P(1..n)$.

OR-composition: $R^x = \{r | r \in \bigcup_{\forall l \in \mathrm{Lic}} R^l\}$;

AND-composition: $R^x = \{r | r \in \bigsqcap_{\forall l \in \mathrm{Lic}} R^l\}$.

Obligation Definite Provability.

$+\Delta^{\mathrm{O}^{l_c}}$: If $P(n+1) = +\Delta^{\mathrm{O}^{l_c}} q$ then,
(1) $\mathrm{O}q \in F$ or
(2) $\exists r \in R^x_s[q] :\ \forall a, Xb, \neg Yd \in A(r)$,
$+\Delta^{l_c} a$, $+\Delta^{X^{l_c}} b$, $-\Delta^{Y^{l_c}} d \in P(1..n)$.

$-\Delta^{\mathrm{O}^{l_c}}$: If $P(n+1) = -\Delta^{\mathrm{O}^{l_c}} q$ then
(1) $\mathrm{O}q \notin F$ and
(2) $\forall r \in R^x_s[q]$:
$\exists a \in A(r)$ or $\exists Xb \in A(r)$ or $\exists \neg Yd \in A(r)$,
$-\Delta^{l_c} a$, $-\Delta^{X^{l_c}} b$, $+\Delta^{Y^{l_c}} d \in P(1..n)$.

OR-composition: $R^x = \{r | r \in \bigcup_{\forall l \in \mathrm{Lic}} R^{\mathrm{O}^l}\}$;

AND-composition: $R^x = \{r | r \in \bigsqcap_{\forall l \in \mathrm{Lic}} R^{\mathrm{O}^l}\}$.

Definite Provability for Prohibitions and Permissions.

Definite proof conditions for prohibitions can be simply obtained from the ones for O.

$\pm\Delta^{\mathrm{F}^{l_c}}$: If $P(n+1) = \pm\Delta^{\mathrm{F}^{l_c}} q$ then
(1)$\pm\Delta^{\mathrm{O}^{l_c}} {\sim} q \in P(1..n)$.

The concept of permission is much more elusive (for a discussion, see, e.g., [5, 20, 22]). Here, we minimize complexities and state that some q is permitted ($\mathrm{P}q$) if it is obtained from an explicit permissive clause for P (i.e., a rule with arrow for $\to^l_{\mathrm{P}}$) supporting q (strong permission of q [25])[25].

$+\Delta^{\mathrm{P}^{l_c}}$: If $P(n+1) = +\Delta^{\mathrm{P}^{l_c}} q$ then
(1) $\mathrm{P}q \in F$ or
(2) $\exists r \in R^x_s[q]$: $\forall a, Xb, \neg Yd \in A(r)$,
$+\Delta^{l_c} a$, $+\Delta^{X^{l_c}} b$, $-\Delta^{Y^{l_c}} d \in P(1..n)$.

$-\Delta^{\mathrm{P}^{l_c}}$: If $P(n+1) = -\Delta^{\mathrm{P}^{l_c}} q$ then
(1) $\mathrm{P}q \notin F$ and
(2) $\forall r \in R^x_s[q]$:
$\exists a \in A(r)$ or $\exists Xb \in A(r)$ or $\exists \neg Yd \in A(r)$,
$-\Delta^{l_c} a$, $-\Delta^{X^{l_c}} b$, $+\Delta^{Y^{l_c}} d \in P(1..n)$.

OR-composition: $R^x = \{r | r \in \bigcup_{\forall l \in \mathrm{Lic}} R^{\mathrm{P}^l}\}$;

AND-composition: $R^x = \{r | r \in \bigsqcap_{\forall l \in \mathrm{Lic}} R^{\mathrm{P}^l}\}$.

3.2.2 Defeasible Provability

As usual in standard DL, to show that a literal q is defeasibly provable we have two choices: (1) We show that q is already definitely provable; or (2) we need to argue using the defeasible part of a multi-license theory D. For this second case, some (sub)conditions must be satisfied. First, we need to consider possible reasoning chains in support of ${\sim}q$ with the modes l_c and X^{l^c}, and show that ${\sim}q$ is not definitely provable with that mode (2.1 below). Second, we require that there must be a strict or defeasible rule with mode at hand for q which can apply (2.2 below). Third, we must consider the set of all rules which are not known to be inapplicable and which permit to get ${\sim}q$ with the mode under consideration (2.3 below). Essentially, each rule s of this kind attacks the conclusion q. To prove q, s must be counterattacked by a rule t for q with the following properties: i) t must be applicable, and ii) t must prevail over s. Thus each attack on the conclusion q must be counterattacked by a stronger rule. In other words, r and the rules t form a team (for q) that defeats the rules s.

Ontology Defeasible Provability.

$+\partial^{l_c}$: If $P(n+1) = +\partial^{l_c} q$ then
(1)$+\Delta^{l_c} q \in P(1..n)$ or
(2) (2.1) $-\Delta^{l_c} {\sim} q \in P(1..n)$ and
(2.2) $\exists r \in R^x_{\mathrm{sd}}[q] : \forall a \in A(r)$: $+\partial^{l_c} a$, and
(2.3) $\forall s \in R^y[{\sim} q]$ either
(2.3.1) $\exists a$: $a \in A(r)$, $-\partial^{l_c} a \in P(1..n)$; or
(2.3.2) $\exists t \in R^x[q]$:
$\forall a \in A(t)$, $+\partial^{l_c} a \in P(1..n)$
and $t \succ s$.

$-\partial^{l_c}$: If $P(n+1) = -\partial^{l_c} q$ then
(1)$-\Delta^{l_c} q \in P(1..n)$ and
(2) (2.1) $+\Delta^{l_c} {\sim} q \in P(1..n)$ or
(2.2) $\forall r \in R^x_{\mathrm{sd}}[q] : \exists a \in A(r)$: $-\partial^{l_c} a$, or

[25] Hence, we do not make explicit in the language that the fact that some p is permitted ($\mathrm{P}q$) can be obtained from the fact that $\neg q$ is not provable as mandatory (weak permission). For a extensive treatment of permission in DL, see [13]

(2.3) $\exists s \in R^y[\sim q]$ such that
(2.3.1) $\forall a$: $a \in A(r)$, $+\partial^{l_c} a \in P(1..n)$; and
(2.3.2) $\forall t \in R^x[q]$:
$\exists a \in A(t)$, $-\partial^{l_c} a \in P(1..n)$
or $t \not\succ s$.

where

OR-composition: (i) $R^x = \{r | r \in \bigcup_{\forall l \in \mathrm{Lic}} R^l\}$, (ii) $R^y = R^x$;

AND-composition: (i) $R^x = \{r | r \in \bigsqcap_{\forall l \in \mathrm{Lic}} R^l\}$, (ii) $R^y = \{r | r \in \bigcup_{\forall l \in \mathrm{Lic}} R^l\}$.

3.2.3 Obligation Defeasible Provability

$+\partial^{\mathrm{O}^{l_c}}$: If $P(n+1) = +\partial^{\mathrm{O}^{l_c}} q$ then
(1)$+\Delta^{\mathrm{O}^{l_c}} q \in P(1..n)$ or
(2) (2.1) $-\Delta^{\mathrm{O}^{l_c}} \sim q \in P(1..n)$ and
(2.2) $\exists r \in R^x_{\mathrm{sd}}[q] : \forall a, Xb, \neg Yd \in A(r)$:
$+\partial^{l_c} a, +\partial^{X^{l_c}} b$,
$-\partial^{Y^{l_c}} d \in P(1..n)$ and
(2.3) $\forall s \in R^y[\sim q]$ either
(2.3.1) $\exists a \in A(s)$ or $Xb \in A(s)$ or $\neg Y \in A(s)$:
$-\partial^{l_c} a \in P(1..n)$, or
$-\partial^{X^{l_c}} b \in P(1..n)$, or
$+\partial^{Y^{l_c}} d \in P(1..n)$; or
(2.3.2) $\exists t \in R^z[q]$:
$\forall a, Xb, \neg Yd \in A(t)$,
$+\partial^{l_c} a, +\partial^{l_c} b, -\partial^{l_c} d \in P(1..n)$, and
$t \succ s$.

$-\partial^{\mathrm{O}^{l_c}}$: If $P(n+1) = -\partial^{\mathrm{O}^{l_c}} q$ then
(1)$-\Delta^{\mathrm{O}^{l_c}} q \in P(1..n)$ and
(2) (2.1) $+\Delta^{\mathrm{O}^{l_c}} \sim q \in P(1..n)$ or
(2.2) $\forall r \in R^x_{\mathrm{sd}}[q] : \exists a \in A(r)$ or $Xb \in A(r)$ or $\neg Yd \in A(r)$:
$-\partial^{l_c} a \in P(1..n)$, or
$-\partial^{X^{l_c}} b \in P(1..n)$, or
$+\partial^{Y^{l_c}} d \in P(1..n)$, or
(2.3) $\exists s \in R^y[\sim q]$ such that
(2.3.1) $\forall a, X^{l_p} b, \neg Y^{l_q} \in A(r)$:
$+\partial^{l_c} a \in P(1..n)$, and
$+\partial^{X^{l_c}} b \in P(1..n)$, and
$-\partial^{Y^{l_c}} d \in P(1..n)$; and
(2.3.2) $\forall t \in R^z[q]$:
$\exists a \in A(t)$ or $Xb \in A(t)$ or $\neg Yd \in A(t)$,
$-\partial^{l_c} a \in P(1..n)$, or
$-\partial^{l_c} b \in P(1..n)$, or
$+\partial^{l_c} d \in P(1..n)$, or
$t \not\succ s$.

where

OR-composition: (i) $R^x = \{r | r \in \bigcup_{\forall l \in \mathrm{Lic}} R^{\mathrm{O}^l}\}$, (ii) $R^y = \{r | r \in \bigcup_{\forall l \in \mathrm{Lic}} R^{\mathrm{O}^l}\} \cup \{r | r \in \bigcup_{\forall l \in \mathrm{Lic}} R^{\mathrm{P}^l}\}$, and (iii) if $s \in R^{\mathrm{O}^l}$, for any $l \in L_c$, then $R^z = \{r | r \in \bigcup_{\forall l \in \mathrm{Lic}} R^{\mathrm{P}^l}\} \cup \{r | r \in \bigcup_{\forall l \in \mathrm{Lic}} R^{\mathrm{O}^l}\}$, otherwise $R^z = \{r | r \in \bigcup_{\forall l \in \mathrm{Lic}} R^{\mathrm{O}^l}\}$;

AND-composition: (i) $R^x = \{r | r \in \bigsqcap_{\forall l \in \mathrm{Lic}} R^{\mathrm{O}^l}\}$, (ii) $R^y = \{r | r \in \bigcup_{\forall l \in \mathrm{Lic}} R^{\mathrm{O}^l}\} \cup \{r | r \in \bigcup_{\forall l \in \mathrm{Lic}} R^{\mathrm{P}^l}\}$, and (iii) if $s \in R^{\mathrm{O}^l}$, for any $l \in L_c$, then $R^z = \{r | r \in \bigsqcap_{\forall l \in \mathrm{Lic}} R^{\mathrm{P}^l}\} \cup \{r | r \in \bigsqcap_{\forall l \in \mathrm{Lic}} R^{\mathrm{O}^l}\}$, otherwise $R^z = \{r | r \in \bigsqcap_{\forall l \in \mathrm{Lic}} R^{\mathrm{O}^l}\}$.

Defeasible Provability for Prohibitions, Permissions. For defeasible prohibitions we have:

$\pm\partial^{\mathrm{F}^l}$: If $P(n+1) = \pm\partial^{\mathrm{F}^l} q$ then
(1)$\pm\partial^{\mathrm{O}^l} \sim q \in P(1..n)$.

$+\partial^{\mathrm{P}^{l_c}}$: If $P(n+1) = +\partial^{\mathrm{P}^{l_c}} q$ then
(1) $+\Delta^{\mathrm{P}^{l_c}} q \in P(1..n)$ or
(2) (2.1) $\mathrm{O}\sim q \notin F$ and $\neg \mathrm{P} q \notin F$ and
(2.2) $\exists r \in R^x_{sd}[q]$ such that
$\forall a, Xb, \neg Yd \in A(r)$:
$+\partial^{l_c} a, +\partial^{X^{l_c}} b$,
$-\partial^{Y^{l_c}} d \in P(1..n)$, and
(2.3) $\forall s \in R^y[\sim q]$, either
(2.3.1) $\exists a \in A(s)$ or $Xb \in A(s)$ or $\neg Yd \in A(s)$:
$-\partial^{l_c} a \in P(1..n)$, or
$-\partial^{X^{l_c}} b \in P(1..n)$, or
$+\partial^{Y^{l_c}} d \in P(1..n)$; or
(2.3.2) $\exists t \in R^z[q]$ such that
$\forall a, Xb, \neg Yd \in A(t)$,
$+\partial^{l_c} a, +\partial^{l_c} b, -\partial^{l_c} d \in P(1..n)$, and
$t \succ s$.

$-\partial^{\mathrm{P}^{l_c}}$: If $P(n+1) = -\partial^{\mathrm{P}^{l_c}} q$ then
(1) $-\Delta^{\mathrm{P}^{l_c}} q \in P(1..n)$ and
(2) (2.1) $\mathrm{O}\sim q \in F$ or $\neg \mathrm{P} q \in F$ or
(2.2) $\forall r \in R^x_{sd}[q]$:
$\exists a \in A(r)$ or $Xb \in A(r)$ or $\neg Yd \in A(r)$:
$-\partial^{l_c} a \in P(1..n)$ or
$-\partial^{X^{l_c}} b \in P(1..n)$ or
$+\partial^{Y^{l_c}} d \in P(1..n)$, or
(2.3) $\exists s \in R^y[\sim q]$ such that
(2.3.1) $\forall a, Xb, \neg Yd \in A(s)$:
$+\partial^{l_c}, +\partial^{X^{l_c}} b$
$-\partial^{Y^{l_c}} d \in P(1..n)$; and
(2.3.2) $\forall t \in R^z[q]$:
$\exists a \in A(r)$ or $Xb \in A(r)$ or $\neg Yd \in A(r)$:
$-\partial^{l_c} a \in P(1..n)$ or
$-\partial^{X^{l_c}} b \in P(1..n)$ or
$+\partial^{X^{l_c}} d \in P(1..n)$, and
$t \succ s$.

where

OR-composition: (i) $R^x = \{r | r \in \bigcup_{\forall l \in \mathrm{Lic}} R^{\mathrm{P}^l}\}$, (ii) $R^y = \{r | r \in \bigcup_{\forall l \in \mathrm{Lic}} R^{\mathrm{O}^l}\}$, and (iii) $R^z = \{r | r \in \bigcup_{\forall l \in \mathrm{Lic}} R^{\mathrm{P}^l}\} \cup \{r | r \in \bigcup_{\forall l \in \mathrm{Lic}} R^{\mathrm{O}^l}\}$;

AND-composition: (i) $R^x = \{r | r \in \bigsqcap_{\forall l \in \mathrm{Lic}} R^{\mathrm{P}^l}\}$, (ii) $R^y = \{r | r \in \bigcup_{\forall l \in \mathrm{Lic}} R^{\mathrm{O}^l}\}$, and (iii) $R^z = \{r | r \in \bigsqcap_{\forall l \in \mathrm{Lic}} R^{\mathrm{P}^l}\} \cup \{r | r \in \bigcup_{\forall l \in \mathrm{Lic}} R^{\mathrm{O}^l}\}$;

3.3 An Abstract Example

Let us consider an example that illustrates some aspects of the proof theory. Assume to work with two licenses l_1 and l_2 and their composition, and let us reason only about obligations and permissions:

$$
\begin{aligned}
F &= \{a, d\}\\
R^{\mathrm{O}^{l_1}} &= \{r_1 : a \Rightarrow^{l_1}_{\mathrm{O}} b,\\
&\qquad r_2 : d \leadsto^{l_1}_{\mathrm{O}} {\sim}e\}\\
R^{\mathrm{P}^{l_1}} &= \{r_3 : \mathrm{O}b \Rightarrow^{l_1}_{\mathrm{P}} d\}\\
R^{\mathrm{O}^{l_2}} &= \{r_4 : a \Rightarrow^{l_2}_{\mathrm{O}} b,\\
&\qquad r_5 : a \Rightarrow^{l_2}_{\mathrm{O}} {\sim}d\}\\
R^{\mathrm{P}^{l_2}} &= \{r_6 : \mathrm{O}b \Rightarrow^{l_1}_{\mathrm{P}} d\}\\
\succ &= \{r_3 \succ r_5\}
\end{aligned}
$$

Let us consider AND-composition heuristics only. A first pair of rules shared by the licenses are $r_1 = r_4$, which lead to $+\partial^{\mathrm{O}^{l_c}} b$. Hence, $r_3 = r_6$ are triggered and can be used to derive another conclusion (see condition (2.2) for $+\partial^{\mathrm{P}^{l_c}}$).

Attacks, however, may come from any rules: in particular, r_5 attacks r_3 and r_6, but r_5 is weaker than r_3, thus we also obtain $+\partial^{\mathrm{P}^{l_c}} d$.

3.4 Properties and Admissibility

The logic presented here is a variant of the one developed in [15, 16]. On account of this fact, two results can be imported here: its soundness and computational feasibility.

THEOREM 3. *Let D be a multi-license theory where the transitive closure of $\succ$ is acyclic. For every $\# \in \{\Delta, \partial\}$, $X \in \{l, Y^l | l \in \mathrm{Lic}, Y \in \{\mathrm{O}, \mathrm{F}\}\}$, and $Z \in \{l, W^l | l \in \mathrm{Lic}, W \in \mathrm{MOD}\}$:*

- *It is not possible that both $D \vdash +\#^Z p$ and $D \vdash -\#^Z p$;*
- *if $D \vdash +\partial^X p$ and $D \vdash +\partial^X {\sim}p$, then $D \vdash +\Delta^X p$ and $D \vdash +\Delta^X {\sim}p$;*
- *if $D \vdash +\partial^{\mathrm{O}} p$ and $D \vdash +\partial^{\mathrm{P}} {\sim}p$, then $D \vdash +\Delta^{\mathrm{O}} p$ and $D \vdash +\Delta^{\mathrm{P}} {\sim}p$.*

PROOF SKETCH. The proof is a trivial variation of the ones for Theorems 1 and 2 in [16]. Theorem 3 shows the soundness of the logic in the sense that it is not possible to derive a tagged conclusion and its opposite, and that we cannot defeasibly prove both p and its complementary unless the definite part of the theory proves them; this means that inconsistency can be derived only if the theory we started with is inconsistent, and even in this case the logic does not collapse to the trivial extensions (i.e., everything is provable). $\square$

Given a multi-license theory D, the *universe* of D (U^D) is the set of all the atoms occurring in D. The *extension* (or conclusions) E^D of D is a structure $(\Delta^+_D, \Delta^-_D, \partial^+_D, \partial^-_D)$, where for $X^l \in \mathrm{MOD}$ and $l \in L$:

$$
\begin{aligned}
\Delta^+_D &= \{Xq : D \vdash +\Delta^{X^l} q\} \cup \{q : D \vdash +\Delta^l q\};\\
\Delta^-_D &= \{Xq : D \vdash -\Delta^{X^l} q\} \cup \{q : D \vdash -\Delta^l q\};\\
\partial^+_D &= \{Xq : D \vdash +\partial^{X^l} q\} \cup \{q : D \vdash +\partial^l q\};\\
\partial^-_D &= \{Xq : D \vdash -\partial^{X^l} q\} \cup \{q : D \vdash -\partial^l q\}.
\end{aligned}
$$

Computational complexity is imported from [15, 16]:

THEOREM 4. *Let*

$$D = (F, L, \{R^l\}_{l \in \mathrm{Lic}}, \{R^{\mathrm{O}^l}\}_{l \in \mathrm{Lic}}, \{R^{\mathrm{P}^l}\}_{l \in \mathrm{Lic}}, \succ)$$

be a multi-license theory. The extension of D can be computed in time linear to the size of the theory, i.e., $O(|\{R^l\}_{l \in \mathrm{Lic}} \cup \{R^{\mathrm{O}^l}\}_{l \in \mathrm{Lic}} \cup \{R^{\mathrm{P}^l}\}_{l \in \mathrm{Lic}}| * |U^D| * |L|)$

PROOF SKETCH. The proof is based on a variation of the data structure used by [16, 15] to prove that a different modal extension of DL has linear complexity. In particular, the logic of [16] consists of counts-as (i.e., ontology) rules, rules for obligations, and rules for agency (capturing successful actions and attempts) labeled by single agents belonging to a finite set of agents. Hence, there is a structural similarity between formal languages (here we have a finite set of licenses labeling modalities, there we have agents). Besides the different conceptual interpretation of the formalism, also the proof theory of [16] is basically the same, except for the fact that actions in [16] are successful (i.e., if agent i sees to it that p, then p is the case) and can be iterated if performed by different agents (e.g., we may have there that agent i sees to it that agent j sees to it that p): but those aspects of the logic of [16] make it more complex with respect to the one developed in this paper. Here, we slightly changed the way in which [16] handles attacks and counterattacks, depending on the heuristics (e.g., AND- or OR-composition) we adopt, and the fact that we also have permissions. However, since permissions are attacked and attack only rules for obligations, it is easy to show that this difference in handling conflicts does not affect the complexity of the logic as the worst case is still when all rules of the theory are considered for detecting and solving any given conflict. $\square$

Finally, let us establish when a license composition l_c is meaningful or admissible. This can be checked taking into account the following guidelines:

- When only defeasible rules and defeaters are considered, a composition is admissible iff it leads to a non-empty set of deontic conclusions. If this set is empty, it means that the composite licenses are fully incompatible from the deontic viewpoint: DL is skeptical logic, so in case there is no way to solve deontic conflicts (according to any given heuristics), it means that the composed license does not produce any effect, and so the composition is deontically meaningless.
- While two defeasible conflicting rules do not lead to any conclusion if we do not know which of them is stronger, in the case of conflicting strict rules there is no way to block contradictory conclusions. Hence, checking if a composition is admissible also requires to exclude that Δ^+_D contains contradictory conclusions.
- Facts are supposed to describe a given situation where licenses are applied, thus they can vary from context to context. This has the following implications. Suppose

we have two licenses l_1 and l_2. If l_1 contains $r_1 : a \Rightarrow_{\mathrm{O}}^{l_1} c$, $r_2 : \mathrm{O}c \Rightarrow_{\mathrm{O}}^{l_1} b$ and $r_3 : e \rightsquigarrow_{\mathrm{O}}^{l_1} \neg c$, l_2 contains $r_3 : a, d \Rightarrow_{\mathrm{O}}^{l_1} c$, and l_1 and l_2 share the same set of facts a, d, we do not have any actual conflict and, for instance with the OR-composition heuristics, we derive $\mathrm{O}c$ and $\mathrm{O}b$ from $l_1 \odot l_2$. However, if the facts are a, d, e the unsolvable conflict between r_1/r_3 and r_2 is triggered, thus $l_1 \odot l_2$ does not support any deontic conclusion. Hence, we may have two levels for detecting unsolvable conflicts in the licenses' composition: when we consider specific sets of facts, or when we examine licenses in general.

The following definition formally considers all these aspects:

Definition 5. *Let*

$$D = (F, L, \{R^l\}_{l \in \mathrm{Lic}}, \{R^{\mathrm{O}^l}\}_{l \in \mathrm{Lic}}, \{R^{\mathrm{P}^l}\}_{l \in \mathrm{Lic}}, \succ)$$

*be a multi-license theory and $\mathcal{A}_D$ is the set of all literals and modal literals occurring in the antecedent of all rules of D. The composite license $l_c = l_1 \odot \cdots \odot l_n$ is F-*admissible *iff*

- $L = \{l_1, \ldots, l_n\}$,
- $\exists Xq \in \partial_D^+$, *and*
- *for any literal p, if $X \in \{l, Y^l | l \in \mathrm{Lic}, Y \in \{\mathrm{O}, \mathrm{F}\}\}$, and $Z \in \{l, W^l | l \in \mathrm{Lic}, W \in \mathrm{MOD}\}$, then we do not have (i) $D \vdash +\Delta^X p$ and $D \vdash +\Delta^X {\sim}p$, and (ii) $D \vdash +\Delta^{\mathrm{O}} p$ and $D \vdash +\Delta^{\mathrm{P}} {\sim}p$.*

The composite license $l_c = l_1 \odot \cdots \odot l_n$ is admissible *iff it is F-admissible for all $F \subseteq \mathcal{A}_D$.*

3.5 Constraining-value Heuristics

This heuristics suggests to add to the composite license the "most constraining clause" in a set of clauses, i.e., the one whose compliance entails the compliance of others in the set. This idea is required in this context to devise a formal account of concept inclusion[26].

Definition 6. *Let $D = (F, L, \{R^l\}_{l \in \mathrm{Lic}}, \{R^{\mathrm{O}^l}\}_{l \in \mathrm{Lic}}, \succ)$ a multi-license theory. If $p, q \in \mathrm{Lit}$, we write $p \sqsubseteq_D q$ to say that the literal p is conceptually included by q in D. If $\mathcal{P} = \{r | r \in \bigcup_{l \in \mathrm{Lic}} R^l, p \in A(r)$ or $p \in C(r)\}$, $p \sqsubseteq_D q$ if*

- $\exists l \in L : D \vdash +\partial^l p$ *and* $D \vdash +\partial^l q$, *and*
- $D_{\sqsubseteq} = (F_{\sqsubseteq}, L, \{R^l_{\sqsubseteq}\}_{l \in \mathrm{Lic}}, \{R^{\mathrm{O}^l}\}_{l \in \mathrm{Lic}}, \succ) \not\vdash +\partial^l q$ *where*

  $F_{\sqsubseteq} = F \setminus \{p, q\}$
  $\{R^l_{\sqsubseteq}\}^{\sqsubseteq}_{l \in \mathrm{Lic}}$ *is such that* $\{r | r \in \bigcup_{l \in \mathrm{Lic}} R^l_{\sqsubseteq}\} \setminus \mathcal{P}$.

In other words, a literal p is included by q in a theory D when both are obtained in D using ontology rules, but, if p and q are removed from the set of facts (if they are there) and all ontology rules having p in the head of body are removed as well, then q no longer follows from the theory. Hence, it means that p is decisive for having q. For space reasons, let us illustrate how to implement the heuristics in the proof theory focusing only on positive definite conclusions:

$+\Delta^{\mathrm{O}^{l_c}}$: If $P(n+1) = +\Delta^{\mathrm{O}^{l_c}} q$ then,
(1) $\nexists p \in U^D$: $q \sqsubseteq_D p$ and (2) $\mathrm{O}q \in F$ or
(3) $\exists r \in R^x_s[q]$:
$\forall a, Xb, \neg Yd \in A(r)$,
$+\Delta^{l_c} a$, $+\Delta^{X^{l_c}} b$, $-\Delta^{Y^{l_c}} d \in P(1..n)$.

$+\Delta^{\mathrm{F}^{l_c}}$: If $P(n+1) = +\Delta^{\mathrm{F}^{l_c}} q$ then
(1)$+\Delta^{\mathrm{O}^{l_c}} {\sim}q \in P(1..n)$, and
(2) $\nexists p \in U^D$: $p \sqsubseteq_D q$.

$+\Delta^{\mathrm{P}^{l_c}}$: If $P(n+1) = +\Delta^{\mathrm{P}^{l_c}} q$ then
(1) $\nexists p \in U^D$: $p \sqsubseteq_D q$, and
(2) $\mathrm{P}q \in F$ or
(3) $\exists r \in R^x_s[q]$: $\forall a, Xb, \neg Yd \in A(r)$,
$+\Delta^{l_c} a$, $+\Delta^{X^{l_c}} b$, $-\Delta^{Y^{l_c}} d \in P(1..n)$.

3.6 A Real-life Example

We show now how to use our logic to compose three licenses widely adopted in the Web of Data. The licenses are as follows:

CC-zero : The Creative Commons CC Zero License[27] is a fully open license, where no obligations are specified. The work is dedicated to the public domain by waiving all of the rights to the work worldwide under copyright law. The deontic component of CC-zero is:

- Permissions: Reproduction, Distribution, Derivative.

ODbL : The Open Database License[28] releases the data constrained by some obligations which are Attribution and Share-Alike for Databases. The deontic component of ODbL is:

- Permissions: Sharing, Adaptation.
- Obligations: Attribution, ShareAlike.

CC-BY-NC-ND : The Attribution-NonCommercial-NoDerivs 2.0 Generic License[29] is a restrictive license for data which is definable as *creative work*. The deontic component of CC-BY-NC-ND is:

- Permissions: Sharing.
- Obligations: Attribution.
- Prohibitions: NonCommercial, NoDerivatives.

Let us consider the following scenario for licenses composition: three interlinked datasets queried by the consumer are released respectively under the CC-zero, ODbL, and CC-BY-NC-ND licenses. The data provider aims at licenses composition such that the composite license returned together with the data provides the wider set of permissions and the smallest set of obligations, as envisioned by the Web of Data philosophy. This result can be achieved by simulating the two heuristics we propose, namely AND-composition and OR-composition, possibly combined with the Constraining-value one. In this example, the multi-license theory D is as follows (the $\rightarrow_c$ with no superscript is a shortcut to mean that the rule holds in all licenses):

[26]The problem of concept inclusion is out of the scope of this paper. Hence, to keep the system manageable, we choose Definition 6, which is perhaps the simplest option in DL. More refined options for DL are offered in [14]. In different contexts, advanced techniques for combining Defeasible Reasoning and Descriptions Logics are discussed, e.g., by [26, 7].

[27]`http://creativecommons.org/publicdomain/zero/1.0/`
[28]`http://opendatacommons.org/licenses/odbl/`
[29]`http://creativecommons.org/licenses/by-nc-nd/2.0/`

- Facts: $F = \{KeepOpen\}$
- Licenses: $L = \{l_{CC-zero}, l_{ODbL}, l_{BY-NC-ND}\}$
- Rules: $R = \{r_1 : \Rightarrow_{\mathrm{O}}^{l_{ODbL}} Attribution$
  $r_2 : \Rightarrow_{\mathrm{O}}^{l_{ODbL}} ShareAlike$
  $r_3 : \Rightarrow_{\mathrm{O}}^{l_{BY-NC-ND}} Attribution$
  $r_4 : KeepOpen \Rightarrow_{\mathrm{P}}^{l_{CC-zero}} Reproduction$
  $r_5 : KeepOpen \Rightarrow_{\mathrm{P}}^{l_{CC-zero}} Distribution$
  $r_6 : KeepOpen \Rightarrow_{\mathrm{P}}^{l_{CC-zero}} Derivative$
  $r_7 : \Rightarrow_{\mathrm{P}}^{l_{ODbL}} Sharing$
  $r_8 : \Rightarrow_{\mathrm{P}}^{l_{ODbL}} Adaptation$
  $r_9 : \Rightarrow_{\mathrm{P}}^{l_{BY-NC-ND}} Sharing$
  $r_{10} : \Rightarrow_{\mathrm{O}}^{l_{BY-NC-ND}} \sim Commercial$
  $r_{11} : \Rightarrow_{\mathrm{O}}^{l_{BY-NC-ND}} \sim Derivative$
  $r_{12} : Derivative \rightarrow_c Adaptation\}$
- Priorities on licenses: $l_{ODbL} \succ l_{BY-NC-ND}$

The composite license l_c is composed by the single licenses such that $l_c = l_{CC-zero} \odot l_{ODbL} \odot l_{BY-NC-ND}$. Notice that AND-composition is not admissible, since there is no rule shared by the three licenses, and so no deontic conclusion can be drawn for l_c. OR-composition is admissible: notice that a conflict arises between rule r_6 and rule r_{11}, which cannot be solved, however. The deontic conclusions are: $+\partial^{\mathrm{O}^{l_c}} Attribution$, $+\partial^{\mathrm{O}^{l_c}} ShareAlike$, $+\partial^{\mathrm{P}^{l_c}} Reproduction$, $+\partial^{\mathrm{P}^{l_c}} Distribution$, $+\partial^{\mathrm{P}^{l_c}} Sharing$, $+\partial^{\mathrm{P}^{l_c}} Adaptation$, $+\partial^{\mathrm{F}^{l_c}} \sim Commercial$, $-\partial^{\mathrm{P}^{l_c}} Derivative$, $-\partial^{\mathrm{F}^{l_c}} Commercial$. The machine-readable version is visualized in Figure 4. If the Constraining-value heuristics is combined with OR-composition, rule r_8 cannot be used and so $-\partial^{\mathrm{P}^{l_c}} Adaptation$.

```
@prefix l4lod: http://ns.inria.fr/l4lod/.
@prefix : http://example/licenses.

:licC a l4lod:License;
    l4lod:obliges l4lod:Attribution;
    l4lod:obliges l4lod:ShareAlike;
    l4lod:permits l4lod:Reproduction;
    l4lod:permits l4lod:Distribution;
    l4lod:permits l4lod:Sharing;
    l4lod:permits l4lod:Adaptation;
    l4lod:prohibits l4lod:Commercial.
```

Figure 4: The composite license `licC`.

4. RELATED WORK

The attachment of additional information like rights or licenses to RDF triplets is linked to an active research field. Carroll et al. [6] noted that RDF does not provide mechanisms (apart from statement reification) for talking about graphs and relations between graphs. They introduced Named Graphs in RDF to allow publishers to communicate assertional intent and to sign their assertions.

There are many differences worldwide related to the copyright of data, and not all data is a copyrightable material. Some of the most popular licenses on the Web include CC, GNU Free Documentation, ODC, Science Commons Database Protocol, and Freedom to Research. The ccREL language is the standard recommended by CC for machine-readable expression of copyright licensing terms [1]. Miller et al. [21] propose the Open Data Commons waivers and licenses that try to eliminate or fully license any rights that cover databases and data. The Waiver vocabulary defines properties to use when describing waivers of rights over data and content. Also the Dublin Core vocabulary can be used to define licenses with the class `dc:LicenseDocument` which provides the legal document giving official permission to do something with the resource and the property `dc:license`.

Iannella [18] presents the Open Digital Rights Language (ODRL) that is a language for the Digital Rights Management community for expressing rights information over content. Gangadharan et al. [9] develop the ODRL-S language by extending ODRL to implement the clauses of service licensing. Gangadharan et al. [10] address the issue of service license composition and compatibility analysis. Their framework is based on the syntax and semantics of the ODRL-S language. They specify a matchmaking algorithm which verifies whether two service licenses are compatible. In case of a positive answer, the services can be composed and the framework determines the license of the composite service.

The work of Villata and Gandon [24] is strictly connected to this line of works [10]. However, there are several differences between the two approaches. First, the application scenario is different, service composition on one side and the Web of Data on the other side. These two different scenarios ask for different treatments of the licensing terms, and open different problems. In particular, the subsumption and compatibility rules defined are different, and this affects also the definition of the composite license.

Truong et al. [23] address the issue of analyzing data contracts. The model is based on ODRL-S [9]. Contract analysis leads to the definition of a contract composition where first the comparable contractual terms from the different data contracts are retrieved, and second an evaluation of the new contractual terms for the data mashup applying composition rules is addressed. This work concentrates on data contracts and not on data licenses, thus there are several differences among the two frameworks. However, there are also some common points like the definition of proper composition rules for merging the clauses of the different licenses/contracts, and the use of RDF for the representation of data licenses/contracts.

Krotzsch and Speiser [19] present a semantic framework for evaluating ShareAlike recursive statements. In particular, they develop a general policy modelling language for supporting self-referential policies as expressed by Creative Commons. The policy language is then instantiated with OWL DL and Datalog. In this paper, we address another kind of problem that is the composition of the licenses into a unique composite license using deontic logic to study the semantics of the composition heuristics.

Gordon [11] presents a legal prototype for analyzing open source licenses compatibility using the Carneades argumentation system. The problem of licenses compatibility is addressed at a different granularity with respect to the purpose of the present paper, and licenses composition is not considered in the prototype. Moreover, in this paper we propose a defeasible logic for both checking the consistency of a set of licenses and combining them into a unique license, and Carneades has been shown to be closely related to defeasible logic [12].

5. CONCLUDING REMARKS

Associating the licensing terms to the data is an open problem in the Web of Data since the absence of automated processing of the licenses prevents data reuse and integration. In this paper, we propose a formal framework to verify the compatibility of a set of licenses, and compose them into a composite license. In particular, we define a deontic logic semantics which allows us to reason over the conflicts among the obligation rules. Three licenses composition heuristics are proposed and formalized. The logic verifies whether some conflicts among the rules arise, and the result of the proof theory is the deontic components of the composite license. The heuristics are used to simulate the generation of the composite license to allow the data provider to guide licenses' composition to obtain the intended behavior, e.g., the composite license maximizes permissions and minimizes obligations and prohibitions.

There are several lines to pursue as future research. First, the quantitative evaluation of the proposed approach is a required step towards the real adoption of the proposed framework in the Web of Data. Second, we still have to consider the case of data obtained by inference from one or several licensed datasets. In particular, a special case we have to address is the one of queries going beyond basic `SELECT` queries, where aggregations are present, e.g., return the *average*, *sum*, etc. of the data possibly over several datasets. Third, the logic should take into account the temporal terms of the licenses.

References

[1] H. Abelson, B. Adida, M. Linksvayer, and N. Yergler. ccREL: The creative commons rights expression language. Technical report, 2008.

[2] C. Bizer, T. Heath, and T. Berners-Lee. Linked data - the story so far. *Int. J. Semantic Web Inf. Syst.*, 5(3):1–22, 2009.

[3] G. Boella, G. Governatori, A. Rotolo, and L. van der Torre. Lex minus dixit quam voluit, lex magis dixit quam voluit: A formal study on legal compliance and interpretation. In *Proceedings of AICOL*, pages 162–183. Springer, 2010.

[4] G. Boella, G. Governatori, A. Rotolo, and L. van der Torre. A logical understanding of legal interpretation. In *Proceedings of KR*. AAAI Press, 2010.

[5] G. Boella and L. van der Torre. Permissions and obligations in hierarchical normative systems. In *Proceedings of ICAIL*, pages 109–118. ACM, 2003.

[6] J. J. Carroll, C. Bizer, P. J. Hayes, and P. Stickler. Named graphs. *J. Web Sem.*, 3(4):247–267, 2005.

[7] G. Casini and U. Straccia. Defeasible inheritance-based description logics. In *Proceedings of IJCAI*, pages 813–818, 2011.

[8] M. Comerio. *Web Service Contracts: Specification, Selection and Composition*. PhD thesis, University of Milano-Bicocca, 2009.

[9] G. R. Gangadharan, V. D'Andrea, R. Iannella, and M. Weiss. Odrl service licensing profile (ODRL-S). In *Proceedings of Virtual Goods*, 2007.

[10] G. R. Gangadharan, M. Weiss, V. D'Andrea, and R. Iannella. Service license composition and compatibility analysis. In *Proceedings of ICSOC, LNCS 4749*, pages 257–269. Springer, 2007.

[11] T. F. Gordon. Analyzing open source license compatibility issues with Carneades. In *Proceedings of ICAIL*, pages 51–55. ACM, 2011.

[12] G. Governatori. On the relationship between carneades and defeasible logic. In *Proceedings of ICAIL*, pages 31–40. ACM, 2011.

[13] G. Governatori, F. Olivieri, A. Rotolo, and S. Scannapieco. Three concepts of defeasible permission. In *Proceedings of JURIX*, pages 63–72. IOS Press, 2011.

[14] G. Governatori, F. Olivieri, A. Rotolo, and S. Scannapieco. Narrowing legal concepts. In *Proceedings of JURIX*, pages 51–60. IOS Press, 2012.

[15] G. Governatori and A. Rotolo. BIO logical agents: Norms, beliefs, intentions in defeasible logic. *Autonomous Agents and Multi-Agent Systems*, 17(1):36–69, 2008.

[16] G. Governatori and A. Rotolo. A computational framework for institutional agency. *Artif. Intell. Law*, 16(1):25–52, 2008.

[17] T. Heath and C. Bizer. *Linked Data: Evolving the Web into a Global Data Space*. Morgan & Claypool, 2011.

[18] R. Iannella. *Open Digital Rights Language (ODRL)*, 2002. http://odrl.net/1.1/ODRL-11.pdf.

[19] M. Krötzsch and S. Speiser. ShareAlike Your Data: Self-referential Usage Policies for the Semantic Web. In *Proceedings of ISWC, LNCS 7031*, pages 354–369. Springer, 2011.

[20] D. Makinson and L. van der Torre. Permission from an input/output perspective. *Journal of Philosophical Logic*, 32(4):391–416, 2003.

[21] P. Miller, R. Styles, and T. Heath. Open data commons, a license for open data. In *Proceedings of LDOW*, 2008.

[22] A. Stolpe. A theory of permission based on the notion of derogation. *J. Applied Logic*, 8(1):97–113, 2010.

[23] H. L. Truong, G. R. Gangadharan, M. Comerio, S. Dustdar, and F. D. Paoli. On analyzing and developing data contracts in cloud-based data marketplaces. In *Proceedings of APSCC, IEEE*, pages 174–181, 2011.

[24] S. Villata and F. Gandon. Licenses compatibility and composition in the web of data. In *Proceedings of COLD*, CEUR Workshop Proceedings 905, 2012.

[25] G. von Wright. *Norm and action: A logical inquiry*. Routledge and Kegan Paul, 1963.

[26] K. Wang, D. Billington, J. Blee, and G. Antoniou. Combining description logic and defeasible logic for the semantic web. In *Proceedings of RuleML*, pages 170–181, 2004.