Towards Licenses Compatibility and Composition in the Web of Data

Serena Villata* and Fabien Gandon

INRIA Sophia Antipolis, France
{firstname.lastname}@inria.fr

Abstract. We propose a general framework to attach the licensing terms to the data where the compatibility of the licensing terms concerning the data affected by a query is verified, and, if compatible, the licenses are combined into a composite license. The framework returns the composite license as licensing term about the data resulting from the query.

1 Introduction

The absence of clarity concerning the licensing terms does not encourage the reuse of the data in the Web of Data [3]. When consumers query the Web of Data, results from different datasets, and thus released under different licensing terms, are provided. In this paper, we propose first to verify the compatibility among the licensing terms associated to a query result, and second, to compose, if compatible, the distinct licensing terms for creating a composite license. The composite license is returned together with the query result using the standard SPARQL query results XML format by means of the <link> element. We adopt Semantic Web languages only, and reuse the Creative Commons (CC) licenses schema [1] to define the anatomy of our licenses. Licenses are composed by models: cc:Permission, cc:Requirement, and cc:Prohibition. Models are composed by elements $el_i$ like ShareAlike, Attribution, and many others. We choose CC because it provides a general schema for licenses specification, even if there are works which should not be released under the CC licenses [5,3].

For addressing this issue and covering a wider range of machine-readable license specifications, we align the CC vocabulary with the other schemas including licensing terms (Figure 1). We extend and adapt existing proposals for licenses compatibility and composition in the area of service license analysis [2] to the Web of Data scenario. However, the different application scenarios (service composition vs Web of Data) open different problems. The compatibility rules we define are different, and the definition of the composite license mirrors the same differences. Truong et al. [6] address the issue of analyzing data contracts using RDF for the contracts representation. This work concentrates on data contracts and not on data licenses. Krotzsch and Speiser [4] present a semantic framework for evaluating ShareAlike recursive statements while we address the problem of licenses composition.

* The author acknowledges the support of the DataLift Project ANR-10-CORD-09.

1 http://www.w3.org/TR/rdf-sparql-XMLres/#head
2 Our proposal

Licenses Compatibility. We define a set of compatibility rules assessing the possible compatibility among the elements composing the licenses $\mathcal{L}$, following [2] for service licenses. First, there are certain elements which are broader in scope of permission than other elements, e.g., Sharing is more permissive than Reproduction [1]. The subsumption rules\(^2\) for cc:Permission elements, and the way they can be combined are shown in Table 1. Table 2.a shows whether an element $el_1$ is compatible with another element $el_2$, i.e., $el_1 \sqcap el_2$, concerning cc:Permission elements\(^3\). The rationale is that these elements are compatible if there is a subsumption relation between them.

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|}
\hline
Subsumption & More permissive & Less permissive \\
\hline
DerivativeWorks $\sqsupset$ Sharing & DerivativeWorks & Sharing \\
Distribution $\sqsupset$ Reproduction & Distribution & Reproduction \\
DerivativeWorks $\sqsupset$ Reproduction & DerivativeWorks & Reproduction \\
Sharing $\sqsupset$ Reproduction & Sharing & Reproduction \\
DerivativeWorks $\sqsupset$ Distribution & DerivativeWorks & Distribution \\
\hline
\end{tabular}
\caption{Compatibility rules for subsumption relation among cc:Permission elements.}
\end{table}

Second, a possible situation in analyzing license compatibility is that one license $\mathcal{L}_i$ specifies clauses which are not specified by the other license $\mathcal{L}_j$, e.g., $\mathcal{L}_i$ specifies cc:Prohibition and $\mathcal{L}_j$ does not specify this clause. Table 2.b shows the compatibility rules for specified elements against Unspecified elements\(^4\). The requirement for specification of Attribution does not affect the compatibility with Unspecified (the same holds for Notice, SourceCode, and CopyLeft). Concerning prohibitions, these elements are not compatible with Unspecified, e.g., commercial use is the default setting of the licenses, thus we cannot assume compatibility if commercial use is denied by NonCommercial. For permissions, we follow the “conservative” approach where unspecification means a denial of compatibility\(^5\).

\(^2\) Subsumption $\sqsupset$ means that there is a compatibility if a certain license element $el_i$ is more permissive, i.e., it accepts more, than the other license element $el_j$.

\(^3\) Rules are expressed under the form of truth tables where elements are evaluated as compatible $T$, or incompatible $F$.

\(^4\) We interpret unspecified elements as “do not care”, as in [2].

\(^5\) Table 2.c shows three exceptions of compatible elements from distinct models.
Let \( L(C) \) be the set of licenses associated to the named graphs affected by the consumer’s query, we say that two licenses are compatible if the models in both the licenses are compatible \([7]\). The models are compatible if (i) the models are the same, (ii) the models are composed by elements which satisfy the compatibility rules (Table 2.c), and (iii) their elements are compatible. The elements are compatible if (i) the elements are the same, (ii) the elements satisfy the subsumption rules (Table 1), (iii) the elements satisfy the compatibility rules against \( \text{Unspecified} \) (Table 2.b), and (iv) the elements satisfy the compatibility rules (Table 2.a-c). If the licenses are not compatible, then we leave to the data provider to decide the strategy to deal with this situation, e.g., the data is returned together with the more constraining license among \( L(C) \).

**Licenses Composition.** If the licenses are compatible then we compose them such that the resulting composite license \( L_c \) (\( \odot \) is the composition relation) satisfies the following properties: \( L_c \) can be generated only if all the licenses composing it are compatible, and \( L_c \) is consistent with the set of licenses used to compose it. The definition of \( L_c \) is achieved through the definition of (i) redefinition rules to be applied in case a subsumption relation holds (Table 1), (ii) composition rules necessary to maintain the consistency of \( L_c \) w.r.t. \( L(C) \) (Table 2.c), and (iii) heuristics to compose the elements of each license into \( L_c \). We consider three basic heuristics: \textit{OR-composition}: \( \forall l \in L_1 \) then \( l \in L_c \); \textit{AND-composition}: if \( \exists l \in L_1 \land \ldots \land L_n \) then \( l \in L_c \); \textit{Constraining-value}: most constraining \( l \in L(C) \) is included in \( L_c \). We leave to the data provider the choice of her best strategy for composing the licenses, e.g., AND-composition typically leads to a shorter and simpler license, while OR-decomposition leads to a more complex license where all the clauses in \( L(C) \) are listed. For example, assume we want to analyze two licenses \( L_1 \) and \( L_2 \) (Figure 2.a-b). We first compare the two licenses at the model level. Both licenses contains the model \textit{cc:Permission}. Even if the two elements are not the same, there is a subsumption relation between them (Table 1). The first license does not contain the model \textit{cc:Prohibition} and the

### Table 2: (a) Compatibility rules among \textit{cc:Permission} elements, (b) Compatibility rules among \textit{cc:Requirement}, \textit{cc:Prohibition}, \textit{cc:Permission} elements against \textit{Unspecified}, (c) Composition rules among \textit{cc:Requirement} and \textit{cc:Prohibition} elements.
second license does not contain the model cc:Requirement. However, the two elements cc:ShareAlike and cc:CommercialUse are compatible (Table 2.c). Thus, the two licenses are compatible. $L_c$ is obtained using the more permissive permission, and the OR-composition heuristic.

3 Future challenges

The first point to be addressed is a legal and social validation of the proposed framework. This means to evaluate, not only quantitatively the performance of the algorithms to retrieve remote licensing statements referenced in the data, but also the legal value of the composite license. We will address an evaluation of the disparate named graphs that might be used in a “typical” query, and their relative proportions that have compatible or incompatible licenses. Finally, we are defining more complex heuristics like the one looking for the minimal composite license.

References