



CaLi: A Lattice-Based Model for License Classifications

Benjamin Moreau, Patricia Serrano-Alvarado, Emmanuel Desmontils

► To cite this version:

Benjamin Moreau, Patricia Serrano-Alvarado, Emmanuel Desmontils. CaLi: A Lattice-Based Model for License Classifications. 34ème Conférence sur la Gestion de Données – Principes, Technologies et Applications (BDA 2018), Oct 2018, Bucarest, Romania. hal-01934596

HAL Id: hal-01934596

<https://hal.science/hal-01934596>

Submitted on 26 Nov 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

CaLi: A Lattice-Based Model for License Classifications

Benjamin Moreau
Nantes University, LS2N, CNRS
OpenDataSoft
Benjamin.Moreau@ls2n.fr
Benjamin.Moreau@opendatasoft.
com

Patricia Serrano-Alvarado
Nantes University, LS2N, CNRS
Patricia.Serrano-Alvarado@ls2n.fr

Emmanuel Desmontils
Nantes University, LS2N, CNRS
Emmanuel.Desmontils@ls2n.fr

1 INTRODUCTION AND MOTIVATION

Web applications facilitate combining resources (linked data, web services, source code, documents, etc.) to create new ones. For a resource producer, choosing the appropriate license for a combined resource involves choosing a license compliant with all the licenses of combined resources and controlling the reusability of the resulting resource through the compatibility of its license. The risk is either, to choose a license too restrictive making the resource difficult to reuse, or to choose a not enough restrictive license that will not sufficiently protect the resource.

We consider that a license l_j is compliant with a license l_i if a resource licensed under l_i can be licensed under l_j without violating l_i . If a license l_j is compliant with l_i then we consider that l_i is compatible with l_j .

Finding the right trade-off between compliance and compatibility is a difficult process. An automatic classification over licenses protecting resources would facilitate this task.

Our research question is: *given a license l_i , how to automatically position l_i over a set of licenses in terms of compatibility and compliance?*

We propose CaLi, a lattice-based model to classify licenses. CaLi is able to answer questions like, “what are the licenses with which l_i is compliant?” and “what are the licenses with which l_i is compatible?”. We show the usability of a CaLi classification through a prototype of a license-based search engine for the Web of Data.

The remainder of this abstract is as follows. Section 2 discusses related works, Section 3 introduces the CaLi model, Section 4 shows the implementation of a license-based search engine for linked data and Section 5 concludes.

2 RELATED WORK

There exist some tools to facilitate the creation of license compliant resources such as TLDRLegal¹, CC Choose², ChooseALicense³, CC search⁴, Web2rights⁵ and Licentia⁶. Unfortunately, one of them is able to order licenses in terms of compatibility or compliance.

The problem of license compatibility and license combination are addressed in Web services [3], in collaborative environments [6] and in the Web of Data [4, 8, 9]. These works focus on combining operators for automatic license combination but do not propose to position a license over a set of licenses.

Concerning the problem of license classification to facilitate the selection of a license, [1, 5] propose a compatibility graph to classify

licenses. However, such graph is build from a manual interpretation of each license, so its generalization and automation is not possible.

In the domain of access control, [2] proposes a lattice model of secure information flow to automatically classify security classes with associated resources. [7] describes several models based on this approach but none focuses on classifying licenses.

None of these works answers our research question. In our work we propose a lattice-based model inspired by [2]. Existing combining operators like [3, 4, 8, 9] make feasible the automatic generation of a lattice. This model is independent on any license combination approach, license description language and licensed resources so that it can be used in a wide variety of domains.

3 CALI: A LATTICE-BASED LICENSE MODEL

CaLi is a model that allows to express license classifications with a lattice structure. The model considers a finite set of licenses, a combining operator, a compatibility relation over licenses, resources and their association with licenses, and a set of constraints. The combining operator applied to the set of licenses produces a lattice that expresses the compliance and compatibility among licenses. This model is introduced next.

Definition 3.1. $\text{CaLi} = \langle L, \oplus, \rightarrow, R, \mapsto, C \rangle$.

$L = \{l_1, \dots, l_n\}$ is a set of licenses.

\oplus is a license combining operator.

\rightarrow is the *compatible with* relation defined on pairs of licenses.

$R = \{r_1, \dots, r_m\}$ is a set of resources.

\mapsto is the *protected by* relation between R and L .

$C = \langle C_L, C_{\rightarrow} \rangle$ is a finite set of constraints to express *viability*. C_L is a set of constraints over L and C_{\rightarrow} is a set of constraints over \rightarrow in $L \times L$.

A license is considered as a set of actions. Actions used in L are taken from $A = \{a_1, \dots, a_n\}$, a set of actions that can be applied on resources.

\oplus is an associative and commutative binary operator that specifies for any pair of licenses, the combined license that is compliant with both licenses. For licenses l_i and l_k , if $l_i \oplus l_k = l_j$ then l_j is compliant with l_i and l_k .

For licenses l_i and l_j , we write $l_i \rightarrow l_j$ iff l_i is compatible with l_j (or l_j is compliant with l_i). $l_i \rightarrow l_j$ implies $l_i \oplus l_j = l_j$, i.e., l_j overcomes l_i .

We write $r_i \mapsto l_i$ when the resource r_i is protected by l_i . For $r_i \mapsto l_i$, if l_i is compatible with l_j then r_i can be protected by l_j .

In addition we introduce the concepts of viability of a license and viability of a \rightarrow relation.

- A *viable license* is a license that allows the licensed resource to be used. A license l_i is viable iff it respects all C_L constraints.

¹<https://tldrlegal.com/>

²<https://creativecommons.org/choose/>

³<https://choosealicense.com/>

⁴<https://ccsearch.creativecommons.org/>

⁵<http://www.web2rights.com/creativecommons/>

⁶<http://licentia.inria.fr/>

- A viable compatible with relation, $l_i \rightarrow l_j$, is a relation that allows a resource licensed under l_i to be licensed under l_j . A \rightarrow relation between two licenses is viable iff it respects all $C \rightarrow$ constraints.

l_i is compatible with l_j if there exists a path from l_i to l_j where all compatible with relations and all licenses are viable.

We demonstrate that the set of licenses L , the compatible with relation \rightarrow and the combining operator \oplus form a lattice where:

- (1) $\langle L, \rightarrow \rangle$ is a partially ordered set.
- (2) L is finite.
- (3) L has an infimum (or greatest lower bound) I such that $I \rightarrow l_i \forall l_i \in L$.
- (4) \oplus is a totally defined least upper bound operator on L .

4 A LICENSE-BASED LINKED DATA SEARCH ENGINE

We experiment the usefulness of our work with a license-based search engine for the Web of Data. This search engine is based on the following CaLi classification:

$L = \{\langle P, O, Pr \rangle\}$; P, O, Pr are sets of permissions, obligations and prohibitions containing actions $\in A$. A is the set containing the 72 actions considered by ODRL⁷.

\oplus is the license combining operator, $l_i \oplus l_j \equiv \langle P_{l_i} \cap P_{l_j}, O_{l_i} \cup O_{l_j}, Pr_{l_i} \cup Pr_{l_j} \rangle$ where P_{l_i} is the permission set of l_i , O_{l_i} is the obligation set of l_i , and Pr_{l_i} is the prohibition set of l_i .

\rightarrow is the compatible relation stating that $l_i \rightarrow l_j$ iff $P_{l_i} \supseteq P_{l_j}$ and $O_{l_i} \subseteq O_{l_j}$ and $Pr_{l_i} \subseteq Pr_{l_j}$.

$C = \langle C_L, C \rightarrow \rangle$ where $C_L = \{\omega_{L_1}, \omega_{L_2}\}$ and $C \rightarrow = \{\omega_{\rightarrow_1}, \omega_{\rightarrow_2}\}$.

Concerning the set of constraints that characterizes viable licenses and viable \rightarrow relations:

ω_{L_1} specifies that viable licences have sets of permissions, obligations and prohibitions that have to be mutually disjoint.

ω_{L_2} specifies that if an action a_i is prohibited, all the actions included in a_i must not be permitted nor obligated to be a viable license. For example, a license that prohibits *Use* is not viable if it permits or obligates *Commercial Use*.

ω_{\rightarrow_1} specifies that cc:ShareAlike term obligates the distribution of derivative works only under the same license.

ω_{\rightarrow_2} specifies that if cc:DerivativeWorks is prohibited in a license l_i then resources protected by l_i should not be protected by l_j .

The generation of a lattice has a complexity of $O(2^n)$. In our case, $n = 2|A|$ and the number of nodes in the lattice is 2^{144} . The number of nodes grows exponentially with the number of terms resulting in a combinatorial explosion. Generating the complete lattice is not suitable for a real scenario with a large number of terms. Thus we propose algorithms to generate part of the lattice *on the fly*. Our approach is in $O(n)$ where n is the number of existing nodes.

Answering our questions with the CaLi classification is straightforward. Finding all licensed datasets whose licenses are compliant with a particular license l_i means to retrieve datasets protected by l_i and all datasets protected by licenses that are above l_i in the graph. Similarly, finding all licensed datasets whose licenses are compatible with a specific license l_i means to retrieve datasets protected by l_i and all datasets protected by licenses that are below l_i in the graph.

The prototype of this search engine is available at <http://cali.priloo.univ-nantes.fr>⁸.

5 CONCLUSIONS

In this paper we proposed CaLi, a model to express license classifications. The goal of our work is to encourage the publication and reuse of resources in a license compliant web.

CaLi can be used to classify licenses in different contexts where resources need to be reused. It considers a set of licenses, a combining operator, a compatibility relation over the set of licenses and a set of constraints. The combining operator applied automatically to the set of licenses produces a lattice that expresses both compliance and compatibility among licenses. This lattice guarantees a partial order over licenses.

ACKNOWLEDGMENTS

Authors thank Matthieu Perrin for his helpful comments on this work.

REFERENCES

- [1] Enrico Daga, Mathieu d'Aquin, Enrico Motta, and Aldo Gangemi. October 11-15, 2015. A Bottom-up Approach for Licences Classification and Selection. In *Legal Domain And Semantic Web Applications (LeDA-SWAn) in International Semantic Web Conference (ISWC)*, Bethlehem, USA.
- [2] Dorothy E Denning. 1976. A Lattice Model of Secure Information Flow. *Commun. ACM* 19, 5 (1976), 236–243.
- [3] GR Gangadharan, Michael Weiss, Vincenzo D'Andrea, and Renato Iannella. September 17-20, 2007. Service License Composition and Compatibility Analysis. In *International Conference on Service-Oriented Computing (ICSOC)*, Vienna, Austria. 257–269.
- [4] Guido Governatori, Antonino Rotolo, Serena Villata, and Fabien Gandon. October 21-25, 2013. One License to Compose Them All. A Deontic Logic Approach to Data Licensing on the Web of Data. In *International Semantic Web Conference (ISWC)*, Sydney, Australia.
- [5] Georgia M Kapitsaki, Frederik Kramer, and Nikolaos D Tselikas. 2017. Automating the License Compatibility Process in Open Source Software With SPDX. *Journal of Systems and Software* 131 (2017), 386–401.
- [6] Marco Mesiti, Paolo Perlasca, and Stefano Valtolina. August 26-29, 2013. On the Composition of Digital Licenses in Collaborative Environments. In *Conference on Database and Expert Systems Applications (DEXA)*, Prague, Czech Republic.
- [7] Ravi S. Sandhu. 1993. Lattice-Based Access Control Models. *Computer* 26, 11 (1993), 9–19.
- [8] Valeria Soto-Mendoza, Patricia Serrano-Alvarado, Emmanuel Desmontils, and Jose Antonio Garcia-Macias. October 11-15, 2015. Policies Composition Based on Data Usage Context. In *Consuming Linked Data (COLD) in International Semantic Web Conference (ISWC)*, Bethlehem, USA.
- [9] Serena Villata and Fabien Gandon. November 11-15, 2012. Licenses Compatibility and Composition in the Web of Data. In *Consuming Linked Data (COLD) in International Semantic Web Conference (ISWC)*, Boston, USA, Vol. 905.

⁷<https://www.w3.org/TR/odrl-vocab/#actionConcepts>

⁸A video demonstration is available at <https://youtu.be/YkSWHSiD-Ps>.