DALICC: A Framework for Publishing and Consuming Data Assets Legally

Giray Havur[1,2], Simon Steyskal[1,2], Oleksandra Panasiuk[3], Anna Fensel[3], Victor Mireles[4], Tassilo Pellegrini[5], Thomas Thurner[4], Axel Polleres[1], and Sabrina Kirrane[1]

[1] Vienna University of Economics and Business, Austria
[2] Siemens AG Österreich, Austria
[3] STI Innsbruck, University of Innsbruck, Austria
[4] The Semantic Web Company, Austria
[5] St. Pölten University of Applied Sciences, Austria

Abstract. In this paper we introduce the Data Licenses Clearance Center, which provides a library of machine readable standard licenses and allows users to compose arbitrary licenses. In addition, the system supports the clearance of rights issues by providing users with information about the equivalence, similarity and compatibility of licenses. A beta version of the system is available at https://www.dalicc.net/.

1 The Data Licenses Clearance Center Framework

DALICC stands for Data Licenses Clearance Center. It is a software framework that supports legal experts, innovation managers and application developers in the legally secure reutilization of third party digital assets such as data sets, software or content. The DALICC framework enables the automated clearance of rights, thus helping to detect licensing conflicts and significantly reducing the costs of rights clearance in the creation of derivative works. This is necessary insofar as modern IT applications increasingly retrieve, store and process data assets from a variety of sources. This can raise questions about the compatibility of licenses and the application's compliance with existing law. In order to provide commercial products and services on top of third party data assets, license clearance is necessary to assure legal compatibility[2].

The DALICC framework consists of three main functional components, namely: *license library*, *license search*, and *license composer*, as shown in Figure 1. These are backed by storage for licenses and an automatic reasoning engine.

The *license library* is a repository that contains machine-readable and human-readable representations of the licenses, the former as ODRL policies , and the latter as plain text. These are laid out in a UI as shown in Figure 3.

In the case of *license search*, the user defines a set of permissions or prohibitions (cf. Figure 2) which are then matched against existing licenses via a JavaScript triggered SPARQL query and processed by a reasoning mechanism which returns the licenses that are consistent with the given input.

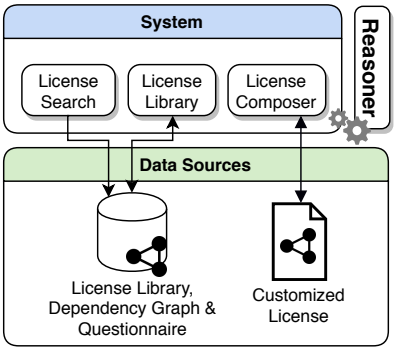

Fig. 1: The Data Licenses Clearance Center Framework

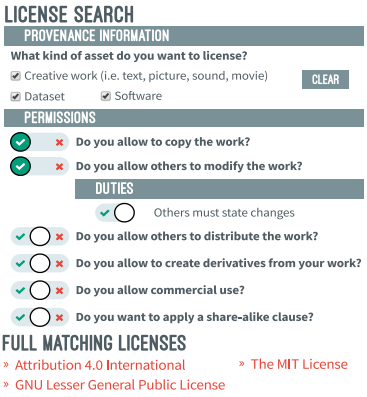

Fig. 2: License search UI

Fig. 3: License library UI

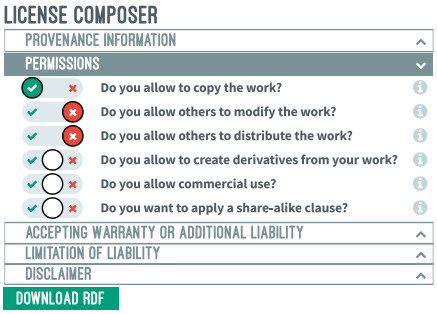

Fig. 4: License composer UI

The *license composer* (cf. Figure 4) allows to create customized licenses from a set of questions which are mapped to ODRL, ccREL and DALICC vocabularies. The composer allows for the declaration of necessary provenance information about an asset (e.g., `purl:title` for the work's title and `cc:deprecatedOn` for the expiration date of the license) and gives the possibility to download an RDF representation and a human-readable version of the created license.

Technology-wise, the DALICC system combines the following components: a Virtuoso[6] triplestore, a Drupal[7] based web application, the PoolParty Semantic Suite[8], and a Clingo Answer Set Programming (ASP) reasoner that checks license consistency and allows to detect conflicts between licenses.

[6] https://virtuoso.openlinksw.com/
[7] https://www.drupal.org
[8] https://www.poolparty.biz/

2 Data Modelling

In order to represent license concepts in a structured machine-readable format we chose the ODRL policy expression language, which includes a flexible and interoperable information model[9] and an extendable vocabulary[10].

The ODRL information model is particularly suitable for modeling licenses in the form of *policies* that express *permissions*, *prohibitions* and *duties* related to the usage of *assets*.

ODRL also defines a vocabulary of general terms (e.g., `odrl:modify` , `odrl:reproduce`, `odrl:distribute`) and can be further extended with terms from other vocabularies such as CC REL (e.g., `cc:CommercialUse`, `cc:DerivativeWorks`)[11] or, like in our case, with a custom one.

To finally model legally valid licenses we extended the expressivity of ODRL with a DALICC vocabulary providing additional legal terms such as `dalicc:worldwide` as a jurisdictional property, `dalicc:perpetual` as a validity type, `dalicc:chargeLicenseFee` as permission and prohibition actions, and `dalicc:modificationNotice` as a duty action.

Additionally, DALICC utilises a dependency graph encoding the expert knowledge about the implicit and explicit semantic dependencies between actions. Following the work of Steyskal and Polleres [3], the dependency graph represents hierarchical relations between actions (e.g., `odrl:sell odrl:includedIn odrl:commercialize`), implications derived from a specific action (e.g., `cc:Attribution odrl:implies cc:Notice`), equalities (e.g., `odrl:copy owl:sameAs odrl:reproduce`), and contradictions between specific actions (e.g., `cc:ShareAlike dalicc:contradicts dalicc:addStatement`).

Figure 5 depicts the central role of `odrl:Action` in integrating the licenses, dependency graph and the composer and search functionalities.

3 Reasoning over Licenses

To reason over licenses we use Answer Set Programming (ASP)[1], a declarative (logic-programming-style) paradigm for solving combinatorial search problems by defining and evaluating rule sets. Licenses are represented in ASP as a set of rules of the form `rule(L,C,I,`α`,T)` where `L`, `C`, `I`, α, and `T` correspond to *license name*, *category of rule*, *assignee*, *action*, and *asset*, respectively.

Policies are derived from the RDF graphs of the licenses. Herein, a rule that permits or prohibits the execution of an action on certain assets does not only affect other rules that govern the execution of the same action on the same asset(s) but also those permitting or prohibiting related actions on the same asset(s). In this sense, CLINGO is an alternative to extensive materialization, which in this case is essential for search, and also enables listing sets of compatible

[9] https://www.w3.org/TR/odrl-model/
[10] https://www.w3.org/TR/odrl-vocab
[11] https://creativecommons.org/ns#

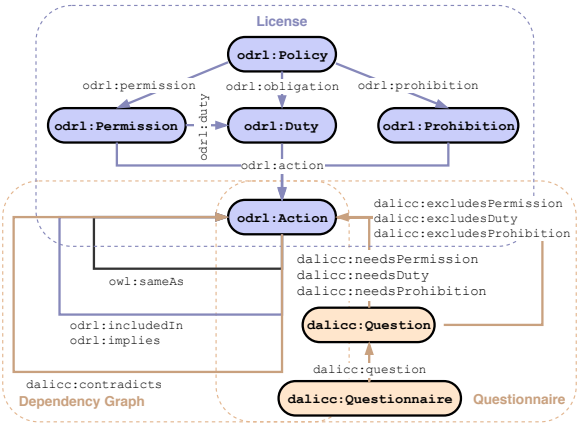

Fig. 5: Interaction between the constituent parts of the framework

statements. This is necessary for effective computation of conflicts between licences, in particular for identifying the conflicting and non-conflicting parts of a license.

4 Conclusion and Future Work

Licensing and rights clearance are complex topics that require a high level of problem awareness and legal expertise. The potential for future work directions are as follows: (i) enabling organizations to create their own applications and workflows using DALICC APIs; (ii) the visualization of data workflows taking into account the license provenance information; (iii) utilizing already existing capabilities of the reasoning component for conflict resolution; (iv) the provision of license management schemes that tackle consistence and trustability issues at the document and workflow level by leveraging transparent infrastructures such as blockchains.

Acknowledgments. Funded by the Austrian Federal Ministry of Transport, Innovation and Technology (BMVIT) DALICC project https://www.dalicc.net.

References

1. Gerhard Brewka, Thomas Eiter, and Mirosław Truszczyński. Answer set programming at a glance. *Communications of the ACM*, 54(12), 2011.
2. Axel Hoffmann, Thomas Schulz, Julia Zirfas, Holger Hoffmann, Alexander Roßnagel, and Jan Marco Leimeister. Legal Compatibility as a Characteristic of Sociotechnical Systems: Goals and Standardized Requirements. *Business & Information Systems Engineering*, 57(2):103–113, April 2015.
3. Simon Steyskal and Axel Polleres. Towards formal semantics for ODRL policies. In *9th International Symposium RuleML*, pages 360–375, 2015.