DALICC As A Service - A Scaleable Architecture for License Clearance

Giray Havur[1,2], Sebastian Neumaier[1] and Tassilo Pellegrini[1]

[1]St. Pölten University of Applied Sciences, Matthias Corvinus-Straße 15, 3100 St. Pölten, Austria
[2]Siemens AG Austria, Technology, Siemensstraße 90, 1210 Vienna, Austria

Abstract

DALICC stands for Data Licenses Clearance Center. It is a software framework that utilizes semantic web standards and linked data principles for the purpose of cost efficient clearing of rights issues in the creation of derivative data and software works. The paper describes the service architecture of and usage scenarios for the DALICC framework, exemplifying a scaleable architecture of semantic web enabled compliance services.

Keywords

license clearance, legal compliance, policy aware system

1. Introduction

Modern IT applications are increasingly composed of various third party components that are provided under various licenses. This can raise questions about the compatibility of licenses and the application‘s compliance with existing law. Manual clearance of licenses can be complex and error-prone, thus requiring a high degree of costly expert knowledge. To lower these costs and improve the quality of license clearance, we developed the DALICC framework [1] that supports the convenient and cost-efficient clearance of licenses in the creation of derivative software and data works by following the semantic web and linked data standards. DALICC can process and reason over RDF representations of licenses, identify conflicts between licenses and support their resolution. While earlier publications on the DALICC framework were mainly concerned with methodological issues of license modelling and reasoning [2], in this paper, we describe the latest developments concerned with the architectural design and associated application scenarios for the user-centric and scalable deployment of the DALICC framework.

2. Related Work

Most of the work on automated processing of licensing information is situated in the context of Rights Expression Languages and contracting [3, 4, 5]. Early work dates back to 1989 and

The Posters and Demos Track of the 17th International Conference on Semantic Systems co-located with the 17th International Conference on Semantic Systems, Amsterdam, Netherlands, September 06–09, 2021

giray.havur@siemens.com (G. Havur); sebastian.neumaier@fhstp.com (S. Neumaier); tassilo.pellegrini@fhstp.com (T. Pellegrini)
0000-0002-6898-6166 (G. Havur); 0000-0002-9804-4882 (S. Neumaier); 0000-0002-0795-0661 (T. Pellegrini)

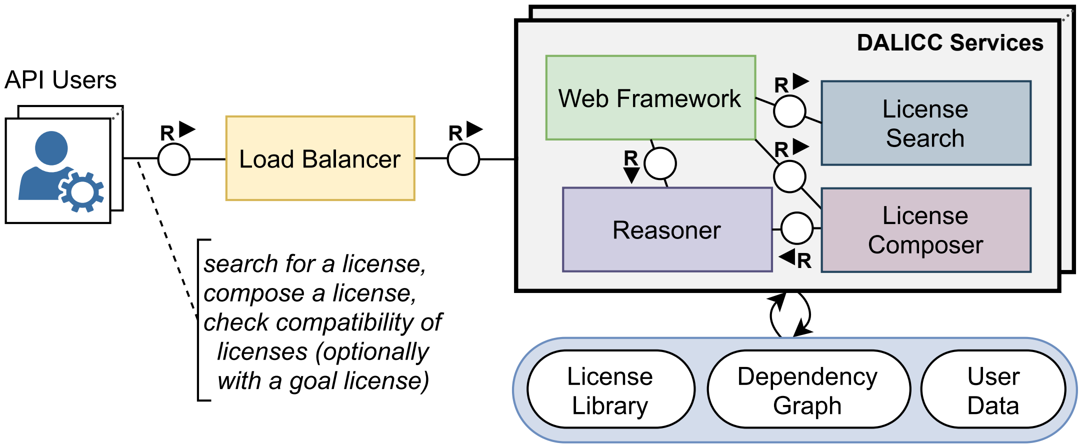

Figure 1: DALICC service architecture.

has since then sparked a rich body of research [6]. One of the first foundational papers on reasoning over licenses was published in 2002 by [7] and extended by [8, 9] towards rule-based resolution of licensing conflicts based on OWL constructs. A proof of applicability was provided by [10] and [11], who describe formalisations of a license clearance tool for derivative works based on the processing of deontic clauses [12, 13]. [10] also provide a demo called Licentia (http://licentia.inria.fr/) that exemplifies the practical value of such a service. More recent work on rule-based conflict detection in license compatibility has been provided by [14]. Our paper contributes to this research by describing a user-centric and scalable architecture that can be understood as a blueprint for semantic web-enabled compliance services.

3. DALICC Service Architecture

The components of the DALICC service framework are shown in Figure 1. An API User's request is first redirected to an available DALICC instance where the web framework delivers the request to the relevant micro-service. To handle multiple API requests, we rely on container-based *load balancing* in front of the services. The *license library* is a repository that contains machine-readable and human-readable representations of the licenses, the former as ODRL policies (set of statements) and the latter as plain text. The *dependency graph* encodes the expert knowledge about the implicit and explicit semantic dependencies between actions [15, 16]. Additionally, we store *user specific data* such as licenses provided by individual users and executed compatibility reports. The following APIs are available:

The **License Search API** queries the license library given specific text or facets.

`/get/licenses/{license_identifier}`: It returns the license definition in a RDF serialization format (e.g., Turtle and JSON-LD) for a given `license identifier`.

`/post/licenses/search/text`: Given a string, it returns a JSON object where license identifiers and descriptions are listed w.r.t. their relatedness to the string.

/post/licenses/search/faceted: Given the asset type (i.e., creative work, dataset, or software) and a list of permissions, duties and prohibitions in an RDF serialization format, it returns a JSON object where license identifiers and descriptions are listed.

The **License Composer API** allows customized licenses to be created from a set of ODRL policy-permission-action-duty statements expressed in an RDF serialization format.

/get/licenses/composer/vocabulary: Returns a JSON object where ODRL, ccREL and DALICC vocabularies for defining license statements are listed.

/post/licenses/composer/check: Returns a JSON object that states if the sent license is formally correct, logically coherent and thus legally valid, and an error message if the license is badly formed.

/post/licenses/composer/upload: Uploads the license in the license library and returns the new license identifier.

The **Compatibility Check API** provides information on equivalence, similarity and compatibility within a set of licenses. It supports two modes: a) reporting all the conflicting pairs of statements in the given set of licenses and b) further checking if a defined goal license subsumes the input set of licenses, e.g., a software developer wants to find out if his/her project can be published under a specific license. In both modes, the reasoner supports conflict resolution by suggesting the minimum removal of statements/licenses for achieving a coherent state.

/post/licenses/compatibility: Given a list of license identifiers and optionally a goal license identifier, it returns the compatibility report.

4. Application Scenarios

In the following section, we provide example application scenarios for the proposed DALICC APIs. Large scale software projects potentially have dozens – or even hundreds – of dependencies to other projects, e.g., to third-party software packages, libraries and modules. Example repositories that list libraries and modules together with their licenses are the Python Package Index ("pip")[1] and the Node Package Manager ("'npm")[2] for the JavaScript programming language. For instance, the popular Python module "pandas" itself already depends on over 80 other modules listed in the respective requirements file.[3]

The goal of the DALICC service is to support programmers and data engineers in the process of creating and publishing new applications that depend on third-party sources. Figure 2 displays how this is envisioned: software projects typically specify a list of dependencies (e.g., a requirements.txt file in Python); initially, the respective licenses get extracted from the listed dependencies and mapped to the DALICC license identifiers. Regarding the identification of licenses, the Software Package Data Exchange[4] (SPDX) provides standardised, short identifiers for a number of standard licenses that potentially serve as canonical permanent URLs. Having

[1] https://pypi.org/

[2] https://www.npmjs.com/

[3] https://github.com/pandas-dev/pandas/blob/master/requirements-dev.txt, last accessed 2021-06-08

[4] https://spdx.org/licenses/, last accessed 2021-06-08

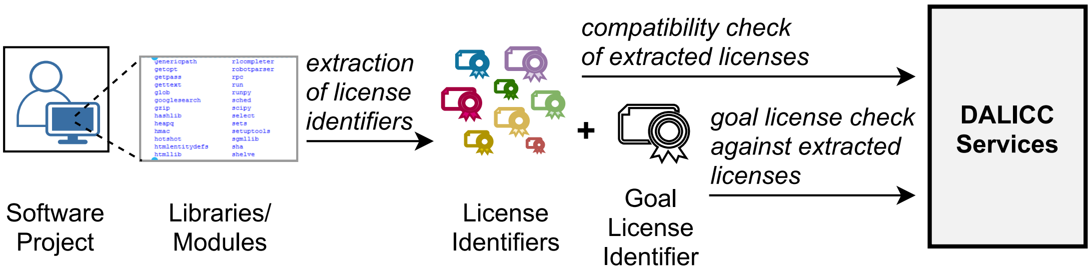

Figure 2: The DALICC services support software projects in the process of (i) checking compatibility of dependencies and (ii) publishing the project under a new license.

the licenses extracted, a programmer can check for compatibility of the dependencies via the DALICC API. If the project gets published under a new license (that exists in the DALICC license library), the API allows to check for compatibility with the specified "goal license" (cf. Figure 2).

5. Concluding remarks

We are currently working on a FAIR documentation and will make the DALICC framework available under a dual license by the end of 2021, thus allowing various forms of collaborative exploitation. The framework closes the existing gap between the technological capabilities to create and publish digital assets and the legal infrastructure necessary to provide them on a legally secure basis for reuse. Hence, DALICC is a tool that puts policies into practice and thus facilitates data governance at various levels. Thus, the DALICC framework should be understood as an enabling service for the emerging data economy.

References

[1] T. Pellegrini, et al., Automated Rights Clearance Using Semantic Web Technologies: The DALICC Framework, in: Semantic Applications, Springer Berlin Heidelberg, Berlin, Heidelberg, 2018, pp. 203–218. URL: http://link.springer.com/10.1007/978-3-662-55433-3_14. doi:10.1007/978-3-662-55433-3_14.

[2] O. Panasiuk, et al., Modeling and Reasoning over Data Licenses, in: A. Gangemi, et al. (Eds.), The Semantic Web: ESWC 2018 Satellite Events, volume 11155, Springer International Publishing, Cham, 2018, pp. 218–222. URL: http://link.springer.com/10.1007/978-3-319-98192-5_41. doi:10.1007/978-3-319-98192-5_41.

[3] V. Rodriguez-Doncel, J. Delgado, A Media Value Chain Ontology for MPEG-21, IEEE Multimedia 16 (2009) 44–51. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6214727. doi:10.1109/MMUL.2009.78.

[4] J. Prenafeta, Protecting Copyright Through Semantic Technology, Publishing Research Quarterly 26 (2010) 249–254. URL: http://link.springer.com/10.1007/s12109-010-9182-3. doi:10.1007/s12109-010-9182-3.

[5] E. Rodriguez, J. Delgado, L. Boch, V. Rodriguez-Doncel, Media Contract Formalization Using a Standardized Contract Expression Language, IEEE MultiMedia 22 (2015) 64–74.

URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6786878. doi:10.1109/MMUL.2014.22.

[6] T. Pellegrini, et al., A Genealogy and Classification of Rights Expression Languages – Preliminary Results, in: Data Protection / LegalTech - Proceedings of the 21st International Legal Informatics Symposium IRIS 2018, Colloquium, Editions Weblaw, Salzburg, Austria, 2018, pp. 243–250.

[7] R. Pucella, V. Weissman, A Logic for Reasoning About Digital Rights, in: Proceedings of the 15th IEEE Workshop on Computer Security Foundations, CSFW '02, IEEE Computer Society, Washington, DC, USA, 2002, pp. 282–294. URL: http://dl.acm.org/citation.cfm?id=794201.795182.

[8] R. García, R. Gil, J. Delgado, Intellectual Property Rights Management Using a Semantic Web Information System, in: D. e. a. Hutchison (Ed.), On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE, volume 3290, Springer Berlin Heidelberg, Berlin, Heidelberg, 2004, pp. 689–704. URL: http://link.springer.com/10.1007/978-3-540-30468-5_44.

[9] R. García, R. Gil, Copyright Licenses Reasoning an OWL-DL Ontology, in: Proceedings of the 2009 Conference on Law, Ontologies and the Semantic Web: Channelling the Legal Information Flood, IOS Press, Amsterdam, The Netherlands, The Netherlands, 2009, pp. 145–162. URL: http://dl.acm.org/citation.cfm?id=1563987.1564000.

[10] R.-D. V&iacute;ctor, V. Serena, G.-P. Asunci&oacute;n, A dataset of RDF licenses, Frontiers in Artificial Intelligence and Applications (2014) 187–188. URL: http://www.medra.org/servlet/aliasResolver?alias=iospressISSNISBN&issn=0922-6389&volume=271&spage=187. doi:10.3233/978-1-61499-468-8-187.

[11] G. Governatori, H.-P. Lam, A. Rotolo, S. Villata, G. A. Atemezing, F. Gandon, LIVE: a tool for checking licenses compatibility between vocabularies and data, 2014. Published: ISWC 2014, 13th International Semantic Web Conference, 19-23 October 2014, Riva del Garde, Italy.

[12] A. Rotolo, S. Villata, F. Gandon, A Deontic Logic Semantics for Licenses Composition in the Web of Data, in: Proceedings of the Fourteenth International Conference on Artificial Intelligence and Law, ICAIL '13, ACM, New York, NY, USA, 2013, pp. 111–120. URL: http://doi.acm.org/10.1145/2514601.2514614. doi:10.1145/2514601.2514614.

[13] E. Cabrio, A. Palmero Aprosio, S. Villata, These Are Your Rights, in: D. e. a. Hutchison (Ed.), The Semantic Web: Trends and Challenges, volume 8465, Springer International Publishing, Cham, 2014, pp. 255–269. URL: http://link.springer.com/10.1007/978-3-319-07443-6_18.

[14] B. Moreau, P. Serrano-Alvarado, M. Perrin, E. Desmontils, Modelling the Compatibility of Licenses, in: e. a. Hitzler (Ed.), The Semantic Web, volume 11503, Springer International Publishing, Cham, 2019, pp. 255–269. URL: http://link.springer.com/10.1007/978-3-030-21348-0_17. doi:10.1007/978-3-030-21348-0_17, series Title: Lecture Notes in Computer Science.

[15] S. Steyskal, A. Polleres, Towards formal semantics for ODRL policies, in: 9th International Symposium RuleML, 2015, pp. 360–375.

[16] T. Pellegrini, et al., Dalicc: A license management framework for digital assets, Internationales Rechtsinformatik Symposion (IRIS) 10 (2019).