This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 870811

SPICE

Social cohesion, Participation, and Inclusion through Cultural Engagement

D4.3: Distributed Privacy and Policy layer: Approach and Implementation (v1.0)

Deliverable information	
WP	WP4
Deliverable dissemination level	PU Public
Deliverable type	R Document, report
Lead beneficiary	OU
Contributors	
Date	31/12/2022
Authors	Jason Carvalho, Chukwudi Uwasomba, and Enrico Daga (OU)
Document status	Final
Document version	v1.0

PAGE INTENTIONALLY BLANK

Project Information

Project Start Date: 1st May 2020
Project Duration: 36 months
Project Website: https://spice-h2020.eu

Project Contacts

Project Coordinator

Silvio Peroni

ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA
Department of Classical Philology and Italian Studies – FICLIT

E-mail: silvio.peroni@unibo.it

Project Scientific Coordinator

Aldo Gangemi

Institute for Cognitive Sciences and Technologies of the Italian National Research Council

E-mail: aldo.gangemi@cnr.it

Project Manager

Adriana Dascultu

ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA
Executive Support Services

E-mail: adriana.dascultu@unibo.it

SPICE Consortium

No.	Short name	Institution name	Country
1	UNIBO	ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA	Italy
2	AALTO	AALTO KORKEAKOULUSAATIO SR	Finland
3	DMH	DESIGNMUSEON SAATIO - STIFTELSEN FOR DESIGN-MUSEET SR	Finland
4	AAU	AALBORG UNIVERSITET	Denmark
5	OU	THE OPEN UNIVERSITY	United Kingdom
6	IMMA	IRISH MUSEUM OF MODERN ART COMPANY	Ireland
7	GVAM	GVAM GUIAS INTERACTIVAS SL	Spain
8	PG	PADAONE GAMES SL	Spain
9	UCM	UNIVERSIDAD COMPLUTENSE DE MADRID	Spain
10	UNITO	UNIVERSITA DEGLI STUDI DI TORINO	Italy
11	FTM	FONDAZIONE TORINO MUSEI	Italy
12	MAIZE	MAIZE SRL (previously CELI SRL)	Italy
13	UH	UNIVERSITY OF HAIFA	Israel
14	CNR	CONSIGLIO NAZIONALE DELLE RICERCHE	Italy

Executive Summary

SPICE is an EU H-2020 project dedicated to research on novel methods for citizen curation of cultural heritage through an ecosystem of tools co-designed by an interdisciplinary team of researchers, technologists, museum curators engagement experts, and user communities. This technical report D4.3 presents the results of Task 2 of Work Package 4: “Distributed Privacy and Policy layer“. The deliverable relates to the SPICE Project objective 3 by developing an approach to giving partner organisations (e.g. museums) meaningful control over their data by expressing fine-grained, user-tailored policies and terms of use and by developing an approach to dealing with privacy violations in user-contributed content. The report illustrates functionalities of the SPICE Linked Data Hub (LDH) related to access control management, visibility and discoverability of assets, and brokering to negotiate access and use. Data managers can detail information on copyright, licensing, and attribution related to any asset managed by the system. The solution incorporates a content monitoring solution that supports data managers in the identification of personal identifiable information (PII), helping them in complying with the General Data Protection Regulation (GDPR).

Document History

Version	Release date	Summary of changes	Author(s) - Institution
v0.1	01/11/2022	Prepared template	Enrico Daga (OU)
v0.2	20/11/2022	Policy management outline	Jason Carvalho (OU)
v0.3	01/12/2022	Policy management diagrams and screenshots	Jason Carvalho (OU)
v0.4	09/12/2022	Status table and policy management content	Jason Carvalho (OU)
v0.5	15/12/2022	Related work, introduction, and conclusions	Enrico Daga (OU)
v0.6	15/12/2022	Privacy monitoring layer	Chukwudi Uwasomba (OU)
v0.7	20/12/2022	Integrating feedback from reviewers	Alba Morales Tirado (OU), Luigi Asprino (UNIBO), Jason Carvalho (OU), Chukwudi Uwasomba (OU), and Enrico Daga (OU)
v1.0	31/12/2022	Submission to EU	Coordinator

Table of contents

1 Introduction

SPICE is an EU H-2020 project dedicated to research on novel methods for citizen curation of cultural heritage through an ecosystem of tools co-designed by an interdisciplinary team of researchers, technologists, museum curators and engagement experts, and user communities. In the SPICE project, we are researching on a linked non-open data management platform that publishes the content of museum archives and supports the development of sophisticated citizen curation applications, including the collection of user-generated content to be integrated within the archives of memory institutions.

This technical report D4.3 presents the results of Task 4.2: Distributed Privacy and Policy layer (M1-M24: Task leader OU). The deliverable relates to the SPICE Project objective 3 by developing an approach to giving partner organisations (e.g. museums) meaningful control over their data by expressing fine-grained, user-tailored policies and terms of use. The report illustrates the functionalities of the SPICE Linked Data Hub (LDH) related to access control management, visibility and discoverability of assets, and brokering to negotiate access and use. Data managers can detail information on copyright, licensing, and attribution related to any asset managed by the system.

The policy management layer chapter (4) describes the development work required to build the foundations of policy and licensing functionality; the design of data structures, data formats, workflows, and storage mechanisms. We also describe the processes of populating these data structures with standard licensing information, processed from external sources. We show how policies are applied at a number of different levels to datasets within the LDH. Chapter 4 also describes the policy negotiation mechanisms that enable dataset users and managers to enter into an automated dialogue for establishing custom terms of use for LDH data resources. Finally, the chapter shows some of the tools that data managers use to manage license allocation and auditing and how layered licenses are resolved, queried, and served via the LDH API.

Furthermore, the Linked Data Hub supports data managers in identifying privacy violations in user-contributed content. A content monitoring solution automatically recognises personal identifiable information (PII), helping data managers in complying with General Data Protection Regulation (GDPR). The monitoring solution is developed using state-of-the-art tools and libraries for natural language processing, which include spaCy, EntityRuler, and Common regular expression. The combination of these tools gave rise to a robust and high computational system capable of detecting fourteen (14) categories of PII. To standardise the severity of PII found for data managers, we designed a severity model and algorithm. We classify PII according to how directly they identify a person or entity. The overall severity score is determined on the basis of the sum of all the points from the severity factors found in a single document. We have also performed extensive testing and obtained encouraging results. We plan for wider accuracy testing on a range of LDH data and expand upon this in the next deliverable.

The content of this report follows and complements SPICE deliverables D4.1 – introducing the Linked Data Hub [1] and D4.2 – incorporating feedback from SPICE use cases [2].

The rest of the deliverable is structured as follows. The next chapter is dedicated to background and related work (Chapter 2). We review relevant literature on metadata management systems, focusing on methods for representing and managing terms of use and data policies, and privacy issues deriving from incorporating user-generated content.

In Chapter 3 we report on the requirements initially introduced in D4.1 [1] and update them in relation to the functionalities presented in this deliverable. Chapter 4 is dedicated to illustrating the policy management layer. In Chapter 5 we describe the content monitoring system and its application in the case of detecting privacy violations. Finally, we conclude the report in Chapter 6.

2 Background and related work

The result of the recent "Stakeholders' survey on a European collaborative cloud for cultural heritage" [3] reports on a shared interest in the development of collaborative infrastructures for cultural heritage organisations. Regarding tools support, the most popular option refers to systems for creating, sharing, and re-using interactive digital content. However, a recent survey on open access policy and practice [4] in the GLAM sector shows how more than 70% of the organisations have data that cannot be published openly on the Web. Our work in the SPICE project aims at tackling this important issue of the sector. In this chapter, we look at background knowledge and reflect on its utility to solve the problem of managing non-open information and its use in citizen curation applications.

Digital rights management means different things in relation to security enforcement and access control (ACL) (for example, as in [5] and formalisation and reasoning over legal knowledge (terms and conditions, licences) [6, 7]. In this section we focus on technologies for the management of legal knowledge (terms and conditions, licences), considering approaches whose aim is to express and reason upon policies in the meaning of licences, and limiting to the approaches designed to work with the WWW's architecture or principles.

Borissova [8], provides a thorough analysis of copyright-related issues raised by cultural heritage digitisation, confirming how the impact of intellectual property is hampering the economic exploitation of cultural heritage, arguing how *"cultural heritage is an essential economic resource which uniqueness is national competitive advantage and as such it should be fully industrially utilised but under intellectual property protection."*. In particular, cultural heritage is a sector in which intellectual property requires the development of *sui generis* rights, beyond standard definitions [8]. Although initiatives such as rightsstatements.org are influencing organisations to review how rights statements are included in the catalogue metadata, digital archives have limited support for rights data management, often confined to one or two metadata fields (e.g. `dcterms:rights`), which textual content often includes all sort of information, including non-pertinent information such as how the asset was acquired [9]. Instead, rights data management has received increasing attention in the Semantic Web research area in recent years. Therefore, we base our survey on [10] (2007) and [11] (2018), limiting to solutions relevant to rights expression, acquisition, and reasoning, and complement it with works published more recently, which cited the mentioned surveys. Kirrane [11] report on a set of high-level tasks relevant to rights data management, that we report with an alignment with some key requirements for citizen curation introduced in SPICE deliverable D4.1 [1] (see Table 2.1).

Our analysis focuses on rights data management, therefore, we do not survey literature about *enforcing* digital rights, excluding techniques to encrypt copyright information in cultural content, for example, DRM approaches, watermarking, and blockchain. In addition, we leave out datasets and domain ontologies, for example, we don't discuss specific work on representing the EU General Data Protection Regulation with RDF/ODRL[1] and only discuss the RDF Licence Database while exploring tools that rely on it. For an overview of the problems of licence compatibility, composition, and propagation, we refer the reader to [12, 13, 14].

The Open Digital Rights Language (ODRL) [6] is designed to support a fine-grained expression of rights statements, based on three main components: *permissions*, *prohibitions*, and *duties*. The Semantic Web community developed a number of solutions dealing with policy reasoning on top of the W3C ODRL Ontology [15]. A first-order logic semantics for ODRL/XML has been proposed and used to determine precisely when a permission is implied by a set of ODRL statements, showing that answering such questions is a decidable NP-hard problem [16]. We restrict our analysis to methods based on the Open Digital Rights Language (ODRL) that we consider superseded legacy technologies such as MPEG-21[2]. The RDF Licence Database is an example of a resource built with ODRL to improve communication and explanation of licenses by means of a dataset of over 100 licenses [17]. ODRL supports several requirements related to the expression and computation of rights. The Linked Data Hub relies on ODRL as

[1] For example, within the W3C Data Privacy Vocabularies and Controls Community Group (DPVCG): https://www.w3.org/community/dpvcg/ (accessed, 9/11/2020).
[2] https://en.wikipedia.org/wiki/MPEG-21

Table 2.1: List of tasks initially introduced in the survey [11].

Task	Description	Relation to WP4 requirements
Policy selection	Select or compose an appropriate policy for an artifact.	11, 12, 14, 18, 24
Policy communication	Disseminate the policy to (potential) consumers.	11, 12, 14, 18, 24
Monitoring	Monitor the use and distribution of the artifact for policy management.	11, 12, 14, 18, 24
Policy enforcement	Put mechanisms in place to enforce compliance with the policy.	11, 12, 14, 18, 24
Policy interpretation	Interpret the implications of the policy in their own context.	n/a
Compatibility testing	Check that the policy is compatible with that of artifacts they are consuming/producing.	n/a
Usage monitoring	Track usage of the artifact for policy compliance	1
Validation	Check that usage of the artifact is compliant with the policy	n/a

the foundational data model.

The problem of license identification and selection is an important one. Tools such as TLDRLegal[3], CC Choose[4], and ChooseALicense[5] help users to browse licences and select the appropriate one for their resources. However, these do not typically rely on a formal representation of the rights. The RDF Licence Database mentioned before is at the base of the tools **Licentia** [18] and **Licence Picker** [19]. Licentia is a Web system that aims at making it easier for users to select a licence to associate with an asset. Specifically, Licentia aims at supporting producers in understanding license terms, in checking the compatibility of a given licence with the aims of the owner and supporting a graphical visualisation.

Licence Picker uses an ontology – LiPio, developed from the RDF Licence Database. LiPio was built applying Formal Concept Analysis (FCA) as a method for clustering licenses with respect to their formal specification and developing a workflow based on a set of questions, designed by curating the clusters produced by FCA. The resulting workflow allows one to reach a decision by answering 3-5 questions, thus reducing the effort in license identification[6] [19]. The approach could be applied to cluster a custom set of ODRL requirements and provide the basis for effective integration with user interfaces for the acquisition of rights information. Applying a similar approach, **CaLi** applies Formal Concept Analysis (FCA) to automatically position a given license over a set of licenses with relation to compatibility and compliance. The system implements the classification technique into a license-based search engine for the Web of Data. Browsing and selecting licenses are core features of the SPICE Linked Data Hub. In the future, we consider incorporating the approach developed in [19] to provide further support to license selection.

License compatibility testing is a crucial feature in data management pipelines for citizen curation. **SPINdle** is the reasoner used by Licentia to perform tasks such as license checking and compatibility [20]. It implements a logic solver able to reason over deontic statements, involving permissions, prohibitions, and duties. The **Data Licenses Clearance Center (DALICC)** supports legal experts, businesses, and developers in the safe reuse of third-party digital assets such as datasets or software [21]. Specifically, DALICC provides support for determining which asset can be shared with whom and under which conditions, thus lowering the burden of rights clearance. The system involves four components: a license library of ready-made licenses, a license composer that reuses existing license

[3] https://tldrlegal.com/
[4] https://creativecommons.org/choose/
[5] https://choosealicense.com/
[6] Licence Picker: http://data.open.ac.uk/licence-picker

policies to create custom ones, a negotiator implementing tasks such as compatibility and conflict resolution, and a license annotator to support the viewing of human-readable rights. In our work, we extend the DALICC catalogue allowing data managers to design custom licenses.

We review related work on privacy policies on the Web. The Usable Privacy Policy Project (UPP)[7] developed a corpus of more than 20k privacy policies from Web sites. The corpus is used to develop semi-automatic approaches for analysing privacy policies including crowdsourcing, natural language processing, and machine learning. Automatic annotating documents about rights can significantly help the fruition of the content by non-experts. **PrivOnto** [22] is a semantic framework for the analysis of privacy policies. The system is one result of UPP that focuses on adopting a combination of natural language processing (NLP), privacy preference modeling, crowdsourcing, and UI design to pragmatically support users in making sense of websites' existing terms and conditions with the aim of empowering users towards more privacy-aware Web surfing. PrivOnto makes use of Semantic Web technologies to support users, researchers, and regulators in the analysis of privacy policies at scale. The European General Data Protection Regulation (GDPR) calls for technical and organisational measures to support its implementation. The **SPECIAL** H2020 project develops a set of tools for supporting data controllers and processors to automatically check if personal data management and distribution respect the duties set forth in the European regulation. SPECIAL includes a policy language to express consent, policies, regulatory obligations, and reasoning systems for automated compliance checking. Those can be used to demonstrate that data processing setup by controllers or processors does not violate the expressed consent of data subjects, and the related business processes satisfy the requirements of GDPR [23].

rightsstatements.org [24] provides twelve standardised rights statements following the Linked Data practice to be used in online cultural heritage, organised in three categories: *In copyright*, *No copyright*, and *Other*. Organisations can use these rights statements to explain to users how online cultural heritage works can be reused. However, the purpose of rightsstatements.org is not to replace licenses and terms of use with an elaborate, machine-readable representation of rights, but to give a simple and standardised way of explaining the key features of licenses. Online digital archives are increasingly adopting rightsstatements.org, although it was observed how the provided options do not fit all the needs of memory institutions [25].

A specific issue for the creation of citizen curation platforms from existing systems and frameworks concerns the lack of dedicated workflows for the type of user-generated content produced by citizen curation processes. Thanks to their architectural modularity and the extendibility of their representation schemas, most systems, especially in the data management area, have in principle the capability to represent user-generated content, but they don't acknowledge this type of content as part of their workflows. In the current situation, any attempt to use these systems to include user-generated content would fall short of creating the appropriate paths for handling them according to the requirements. Current workflows, in fact, mostly rely on a unidirectional path from ingestion to fruition, with user responses not being reintroduced in the system as first-class citizens. The flexibility of access tools provided by most systems, which rely on effective indexing modules, is generally intended for the end user, and it is not available to create curation paths depending on content types and features (in other terms, they are not ready to enable the *scripting* of activities for generating and managing user-generated content). Also, extending the current solutions to create citizen curation systems may not be feasible for small organisations that cannot afford a similar effort (in terms of know-how, costs, staff, and infrastructures) needed to make significant development work on top of their current solutions. We can conclude from the analysis that the role of mediators will be crucial in supporting the heterogeneity and diversity of organisations involved in citizen curation projects, providing specific services (e.g. license clearance or monitoring inappropriate content) and flexible, cost-effective platforms to design and curate interactions, processes, and data.

Semantic web technologies, in particular, could support many of the requirements reviewed so far. For what concerns semantic data management, however, it emerges that little advancements have been made from the agenda settled in 2012 [26]. Specifically, still insufficient efforts have been made for what concern the availability of integrated knowledge systems, the validation of the extended data models provided by each cultural institution, the capability of handling uncertain reasoning, the multilingualism of the exposed cultural repositories, etc. In general, a crucial element to improve would concern the integration of the existing technologies with the standard workflow operation

[7] https://explore.usableprivacy.org/

of Cultural Experts and Information Professionals (IPs) in Libraries, Archives, and Museums (LAMs) [27]. Finally, and consistently with the requirements provided by the citizen curation ecosystem, introduced in SPICE Deliverable D4.1, the crucial issues concerning the ownership, permissions of use, trust, and copyrights issues have only been recently sketched. In this deliverable, we attempt to fill this gap by designing a *policy management layer* on top of the SPICE Linked Data Hub, and a *privacy monitoring system* as an intelligent agent for supporting the analysis of user-generated content integrated into cultural heritage archives. More insight on the state of affairs of cultural heritage systems in relation to citizen curation can be found in the survey article published by the SPICE project team on the ACM Journal of Computing and Cultural Heritage [28].

3 Privacy and policy layer requirements

The requirements for citizen curation platforms were originally introduced in SPICE Deliverable D4.1 [1] and further updated in D4.2 [2]. In Table 3.1, we report the requirements and update on their progress so far, therefore, showing the contribution of this deliverable within the general progress of the work package. In the following chapters, we illustrate the functionalities implemented in the privacy and policy layer of the SPICE Linked Data Hub.

Table 3.1: Requirements for the SPICE Linked Data Hub: progress update.

	Nickname	Role	Action	Target	D4.1 Status	D4.2 Status	D4.3 Status
1	[AnalyseUsage]	custodian/owner	analyse	access and usage of my data	50%	80%	
2	[BackupContent]	data manager	back-up/restore	my data to support recovery in the case of a loss event.	30%		
3	[BrowseIndex]	builder	browse	an index of the resources I have access to	30%	80%	
4	[BrowseMarketplace]	custodian/owner/builder	browse	a marketplace of offers of digital assets	0%		
5	[ControlMetadata]	owner/custodian	control	the metadata production in the ingestion process	0%	100%	
6	[DetectPII]	custodian/builder	detect	personally identifiable information (PII) included in user-generated content	0%	0%	100%
7	[ExpressCopyright]	custodian/owner	express	the copyright associated with digital assets in my collection	30%		
8	[ExpressExemptions]	custodian/owner	express	exemptions and characterize them	0%		
9	[ExpressFees]	owner	express	fees as duties associated to the permissions granted	0%		
10	[ExpressOffers]	owner	express	offers with relation to the assets I own.	50%		
11	[ExpressPermissions]	owner	express	permissions, prohibitions, constraints and duties	30%	30%	100%
12	[ExpressPolicies]	custodian/owner	express	usage policies in relation to my data	0%	0%	100%
13	[ExpressQualityFeatures]	custodian/builder	express	the quality of the asset and their features	0%		
14	[ExpressTimeConstraint]	owner/custodian	express	time limitations to permissions I grant	0%	0%	100%

15	[ExternalAccessData]	builder	access	data from an external application	100%		
16	[FilterSensitiveContent]	custodian/builder	filter	sensitive content for specific target groups	0%		
17	[GrantCheck]	builder/custodian/owner	verify	lawful access to a collection metadata or digital asset	30%		
18	[GrantRecovery]	owner/custodian/builder	view	terms of use granted	0%	0%	100%
19	[InappropriateContent]	custodian/builder	identify/-filter	user-generated content that can be inappropriate	0%	0%	30%
20	[InspectIngestionProcess]	owner/custodian	inspect	the metadata production in the ingestion process	30%	100%	
21	[ManageAccess]	data manager/custodian/owner	manage	access control to the data	100%		
22	[ManageVisibility]	data manager	manage	visibility of my registered data sources	100%		
23	[MonitorAccess]	data manager/custodian/owner	monitor	access to my data	30%	80%	100%
24	[MultipleRightsAspects]	custodian	express	that multiple subjects hold copyrights on different aspects of the digital asset	0%	0%	100%
25	[NegotiateRights]	custodian	negotiate	rights on behalf of the owner	0%	0%	100%
26	[NominateDelegate]	custodian	nominate	an external entity to negotiate rights on behalf of a copyright owner	0%	30%	100%
ls27	[ObtainCredentials]	builder	obtain	credential details (e.g., API Keys) to data	100%		
28	[OrganiseCollections]	custodian/builder	organise	resources I have access to into customized collections	10%		
29	[ProduceLD]	data manager	produce	linked data from existing non-LD resources	100%		
30	[PublishLD]	data manager	publish	linked data with alternative Linked Data vocabularies (Viewpoints)	0%	100%	
31	[ReadData]	builder	read	data from a dataset –e.g., via a (secured) Web API	100%		
32	[RecognisedAuthor]	owner	be_recogr	as author of the picture of the artwork	0%		
33	[RegisterSources]	data manager	register	existing Linked Data sources	0%	50%	
34	[RequestAccess]	builder	request	access to data	0%	0%	100%
35	[RequestPermission]	builder	request	permission to use a digital asset under specific terms	0%	0%	100%

36	[RevokeRights]	owner/custodian	revoke	usage permissions I granted in the past	30%	100%	
37	[SecureStack]	data manager	secure	the content against malicious attacks	100%		
38	[SetupRepository]	data manager	setup	a data repository	100%		
39	[ShareCollections]	custodian/builder	share	my customized collections as linked data	0%		
40	[UploadDataset]	data manager/owner/-custodian	upload	data to my dataset	100%		
41	[UsagePolicyGrant]	owner/custodian	grant	permission to use a digital asset under requested terms	0%	0%	100%
42	[WriteData]	data manager/builder	write	data to a dataset –e.g., via a (secured) Web API	100%		

4 The Linked Data Hub: policy management layer

The policy management layer of the Linked Data Hub is one of the final pieces of significant software development of Work Package 4. It addresses a large section of the original requirements for the Linked Data Hub, providing the policy data structures, workflows, licensing rules, user interface components, and API functions to enable data owners and managers to effectively control the terms of use of their resources.

In this section, we describe the basic license functions that were developed in the very early stages of the LDH, their limitations, and how they have been updated to address the requirements set out in chapter 3. We begin with a description of the development work required to build the foundations of the policy management layer; data structures have been designed and put in place to store policy data and facilitate licensing and policy workflows. We also describe the processes of populating these data structures with standard license information, processed from external sources.

The report then explains how the policy management layer applies policies at a number of different levels and to various resources through the use of license resource and assignee scope. By offering this level of granularity, combined with appropriate user interface tools and the ability to build application-specific custom licenses, the policy management addresses many of the terms of use requirements initially set out for the LDH.

A significant part of the development of this layer of the LDH is based around license and policy negotiation. Workflows and processes have been devised and described below that show how we have addressed this issue, offering dataset owners and dataset users the chance to enter into an automated negotiation dialogue. This dialogue aims to produce a custom set of agreed terms of use for specific data resources.

Finally, we show some of the tools that data owners and managers might use to manage license allocation and auditing and how layered licenses are resolved, queried, and served via the LDH API.

Development of the policy management layer addresses the specific numbered requirements 11, 12, 14, 18, 24, 25, 26, 35 and 41, as set out in chapter 3.

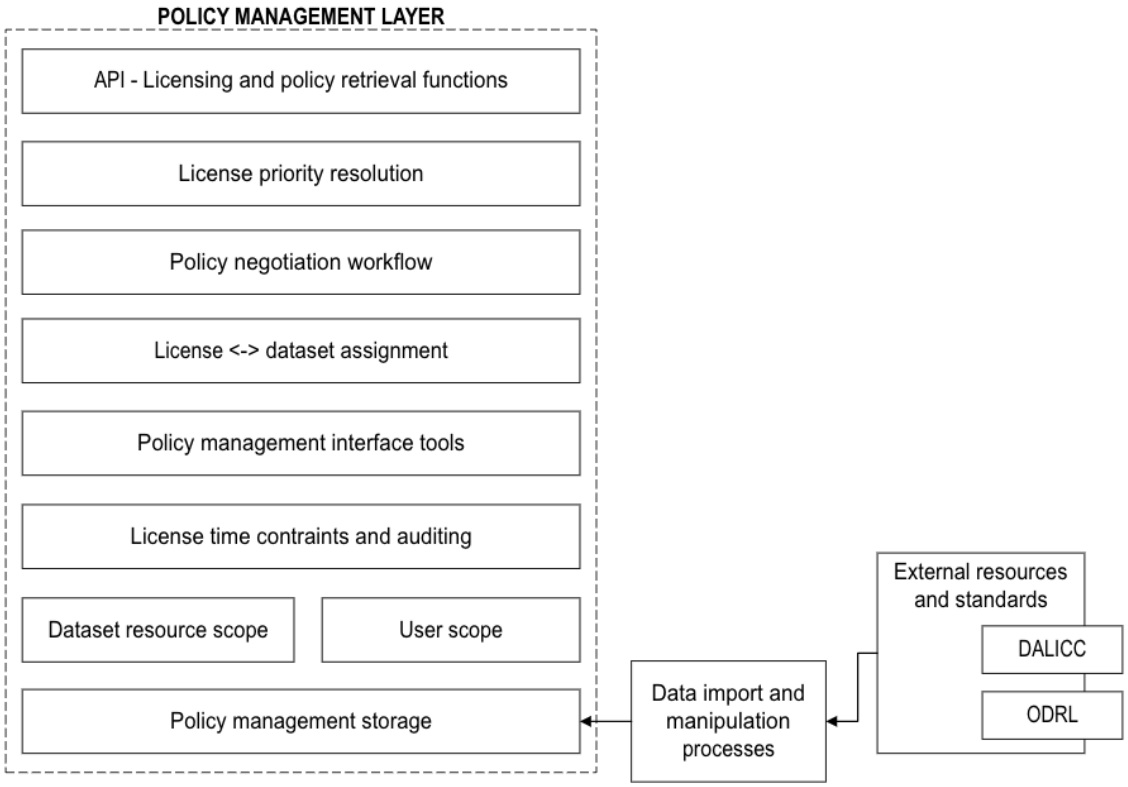

Figure 4.1: Overview of the Linked Data Hub policy management layer

4.1 Original license functions

The previous release of the SPICE LDH contained an Ownership and Licensing tab. This offered dataset owners the facility to populate dataset ownership and attribution details as well as to assign one of five basic licenses to the entire dataset. This facility was offered as an initial basic solution to specifying dataset terms of use. It offered no detailed breakdown of policies for each license, nor the ability to assign licenses at any level of granularity. Licenses were simply comprised of a link to a basic external license description for each entry.

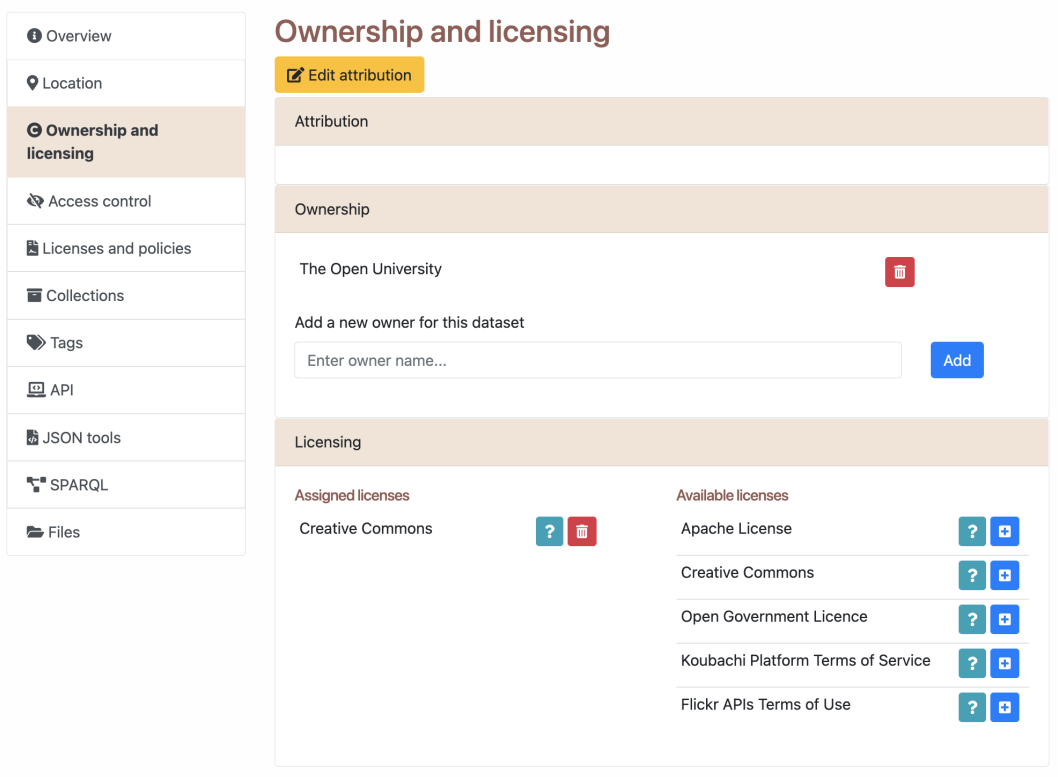

Figure 4.2: Original SPICE LDH Portal licensing features

This dataset tab has now been modified to remove license information, leaving only ownership and attribution details. Licensing and policy details have now been moved to a dedicated tab of their own and are now managed by the SPICE LDH policy management layer. This policy management layer is one of the focuses of this report and provides far more detailed and granular management of dataset license management and uses standards-based license and policy data formats to allow us to make use of existing licensing resources as well as offer the potential for future policy reasoning functionality.

4.2 Making use of existing technologies and methodologies

Where possible, existing technologies and methodologies have been used as a foundation upon which to build the SPICE LDH policy layer. At its core, the SPICE LDH policy layer is built using a library of standard licenses from which dataset owners and managers can select. The licenses are derived from DALICC – The Data Licenses Clearing Center. As part of DALICC's license reasoning software framework, an open API is provided through which a library of standard licenses is made freely available in a machine-readable format.

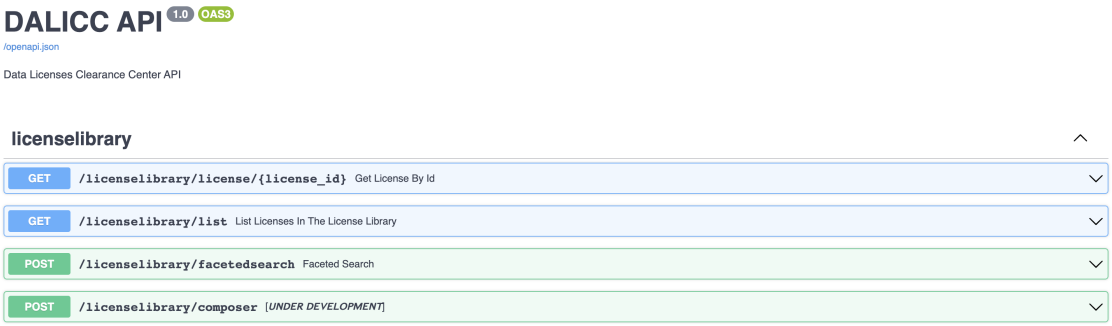

Figure 4.3: DALICC API

Licenses within the SPICE LDH are stored and manipulated using a data format based on Open Digital Rights Language (ODRL) standards. ODRL is a policy expression language that provides a sufficiently flexible and expressive information model, vocabulary, and encoding mechanism for representing licenses and policies within the SPICE LDH.

Licenses are expressed in ODRL as a collection of policies that describe the appropriate use of digital resources. Policies fall into one of three categories; permitted actions, prohibited actions, and any obligations that are required to be met for that digital resource. The ODRL specification describes the use of the policies that make up licenses and how they can be defined. The following diagram shows the full ODRL information model for policies. For this component of SPICE LDH, we are using a subset of this functionality.

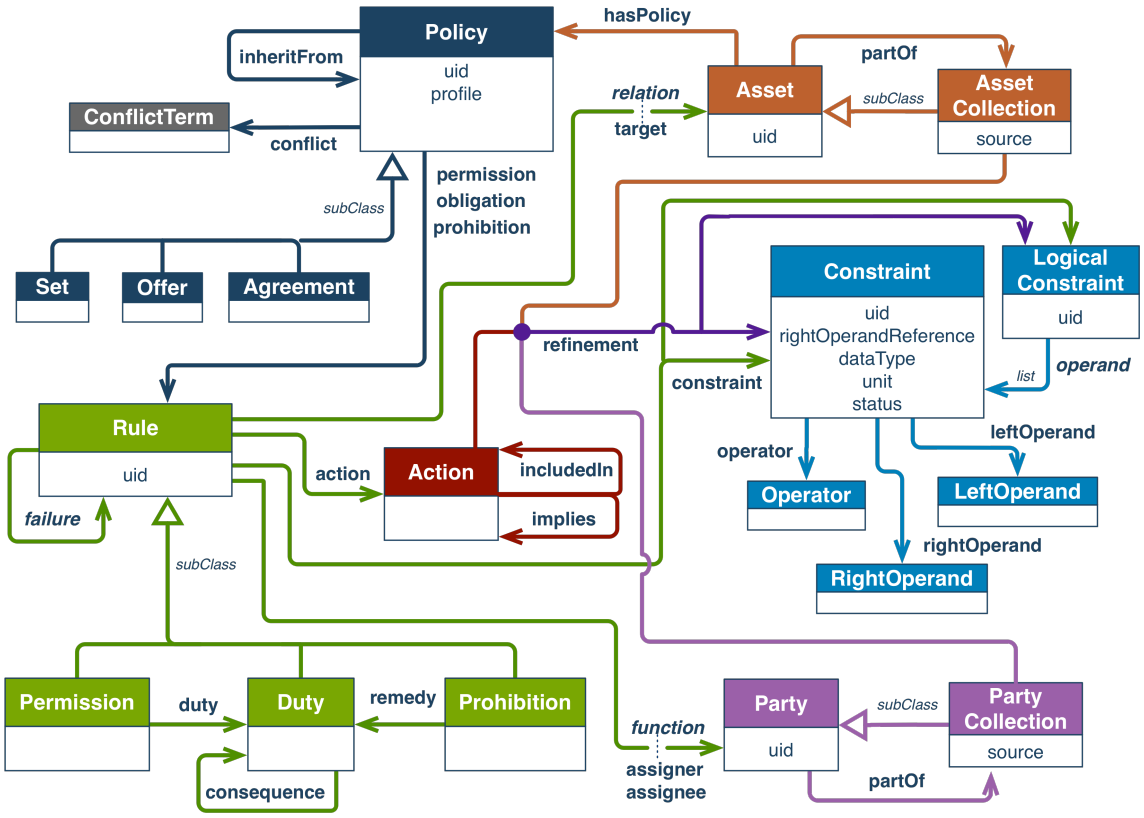

Figure 4.4: ODRL Information Model

4.3 Foundations for policy data management

Before development work could commence on the core policy management tools of the LDH, system design and foundation work was required to put appropriate data and storage mechanisms in place. Building on the work discussed in [29], we devise license and policy storage formats, describe the workflow for importing and storing a library of standard data licenses, and the database structures designed to facilitate license allocation and audit trails of license allocation history.

4.3.1 License storage format - ODRL with extensions

In addition to specific policy descriptions, we extend and build upon the ODRL-compliant attributes to store metadata for each license within our devised data format which enables us to provide a full reference to license sources as well as offer sufficient detail to allow licenses to be applied and manipulated within the SPICE LDH at a number of different levels of granularity. These include licenses assigned to particular users, governing particular dataset sub-resources such as individual JSON documents or files and specifying license validity dates.

```
{
    "@context": [ "https://spice.kmi.open.ac.uk/context/policyLayer.jsonld", "
       http://www.w3.org/ns/odrl.jsonld" ],
    "@type": "odrl:Policy",
    "odrl:uid": "http://example.com/policy:8888",
    "odrl:profile": "http://spice.kmi.open.ac.uk/odrl/profile:1",
    "odrl:target": "http://spice.kmi.open.ac.uk/dataset/dataset-id",
    "odrl:assigner": "http://spice.kmi.open.ac.uk/dataset/dataset-id/admin",
    "odrl:assignee": "http://spice.kmi.open.ac.uk/people/user12345",
    "schema:title": "Licence title",
    "schema:text": "Licence text",
    "active": true,
    "created-time": "12312321323",
    "modified-time": "12345677657",
    "schema:validFrom": "5/7/2022",
    "schema:validUntil": "7/7/2022",
    "odrl:permission": [
        {
            "odrl:action": [
                "odrl:play"
            ]
        },
        {
            "odrl:action": [
                "odrl:distribute",
                "odrl:advertise"
            ],
            "odrl:duty": []
        }
    ],
    "odrl:obligation": [
        {
            "odrl:action": [
                "odrl:distribute",
                "odrl:advertise"
            ]
        }
    ]
}
```

Figure 4.5: Example license using ODRL JSON format

4.3.2 Import process from DALICC

A workflow was devised for collecting license details from the DALICC API and converting them to a suitable format for storage within the SPICE LDH's standard license library. Using SPARQL Anything, a license list, license metadata and individual policy details are retried from the DALICC API in TTL format. The data is then converted to CSV, combined into a list of licenses in JSON format (see figure 4.5) via a Python script, and pushed to the Linked Data Hub via the LDH API.

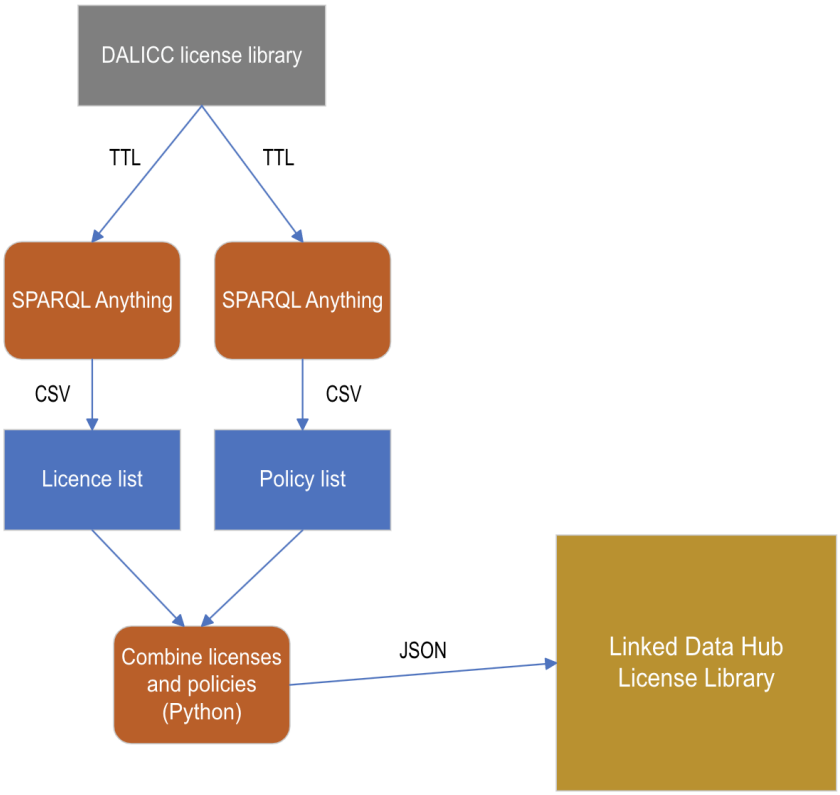

Figure 4.6: License and policy import workflow

4.3.3 Data structures in the LDH

The licensing and policy layer extends the existing dataset metadata data structures to accommodate the storage requirements of these new features. The license and policy functionality makes use of two distinct data structures for each dataset to make the features described in this deliverable possible.

1. **Dataset license library** - Each dataset maintains a list of the licenses allocated to it with information that describes which users the license is accessible by, where appropriate. Within this list, one license with dataset-wide scope will be assigned as the default license, through the use of the `"default":True` attribute within the license JSON object. Where API requests are made with access keys that have not been assigned licenses (for example, legacy keys that were created before the policy layer was implemented) or where the assigned license has since expired or been removed, the default license will be served with the data.
2. **Licenses assigned for key access** - The dataset metadata also maintains a list of the specific licenses that have been associated with specific access keys. Where API data requests are made with keys that are not present in this list, the default license (described above) is served.

In addition to dataset-specific license information, the LDH stores a complete list of the standard licenses in a library (described in 4.3.2) that is made available to all datasets. This is stored centrally within the LDH, outside of dataset-specific metadata.

4.4 License application scope

During the development of the SPICE LDH and subsequent project pilot applications that make use of the LDH technology, it has become apparent that being able to apply licenses and policies at a broad dataset level is prohibitively restrictive. As part of the development of the policy layer for D4.3 and specifically addressing requirement 24 (as detailed in chapter 3), we have expanded the ability to apply licenses at a number of levels within a dataset.

Per user licensing:

Dataset owners should have the ability to assign a license to a particular user, separately from the default license that may be applied to that dataset for all other users. A situation may occur where an individual user or users have negotiated an alternative set of terms for the use of the dataset and the SPICE LDH policy layer should be able to reflect this.

Per dataset resource licensing:

Datasets within the SPICE LDH are made up of a collection of resources; primarily JSON documents and stored files. There may be situations where a named resource within the dataset is required to have a differing set of usage policies associated with it to the default set assigned to the parent dataset. For this reason, the policy layer should offer the ability to assign licenses to individual dataset resources as well as default licenses for the rest of the dataset. These resource-specific policies should also be assignable to individual named users if required.

4.5 License assignees

By default, licenses and policies associated with datasets are applicable to all users of the data within that dataset. There may be circumstances, however, where a particular user is granted the use of data under a set of policies that differs from the default offering. In order to facilitate this, we make use of the ODRL *assignee* attribute, within the JSON object for each stored dataset license. Using this approach, several licenses can be stored alongside a dataset to be made available for selection at the time of key registration (section 4.8).

```
1 {
2     "@context": [ "https://spice.kmi.open.ac.uk/context/policyLayer.jsonld", "
          http://www.w3.org/ns/odrl.jsonld" ],
3     "@type": "odrl:Policy",
4     ...
5     "odrl:assignee": "http://spice.kmi.open.ac.uk/people/user12345",
6     ...
7
8 }
```

Figure 4.7: License assignees stored with each license ODRL JSON object

Note that licenses assigned to named LDH users are not automatically applied to all API requests for data from that users. The *assignee* attribute simply defines which licenses are available to be selected by which users when they register API access keys on LDH data resources.

4.6 License selection interface and tools

Here we show a number of screenshots of the various user interface components that have been added to the LDH portal. These allow dataset owners and managers to inspect, select, remove, and manage the licenses allocated to

their datasets. Through the use of these web-based tools, we address requirements 11, 12, 14, 18, and 41 of this deliverable.

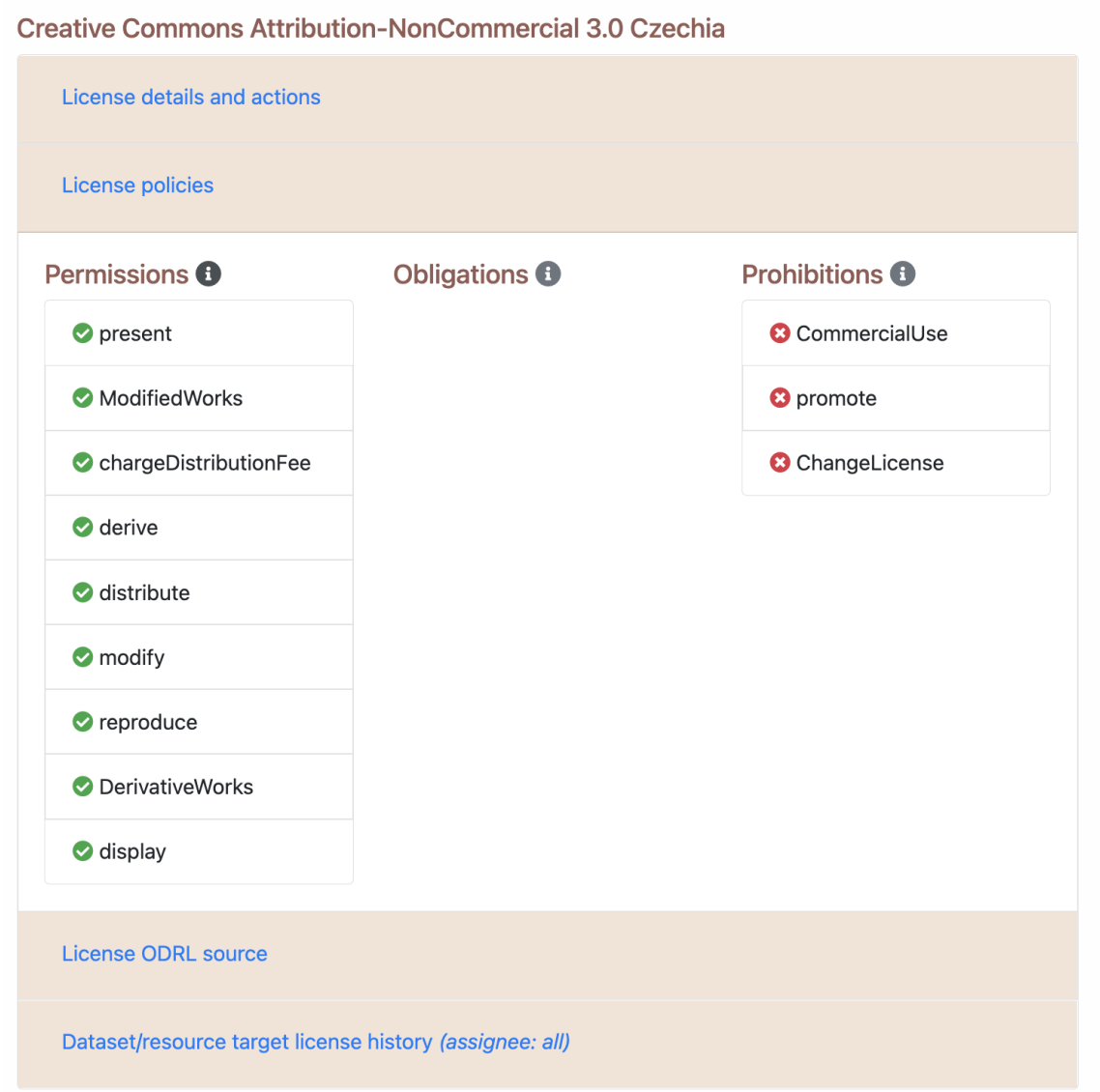

Figure 4.8: View license policies

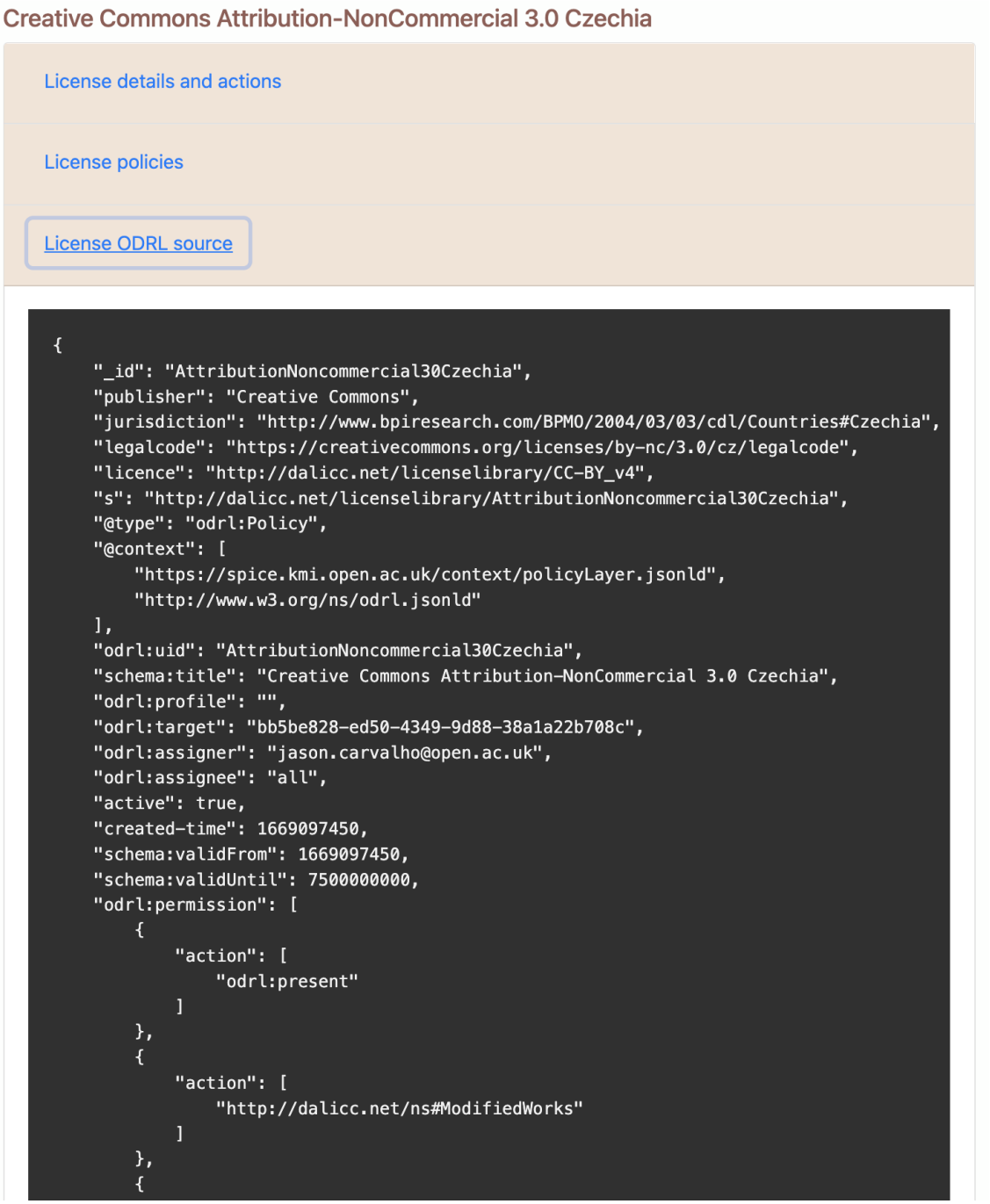

Figure 4.9: View license source

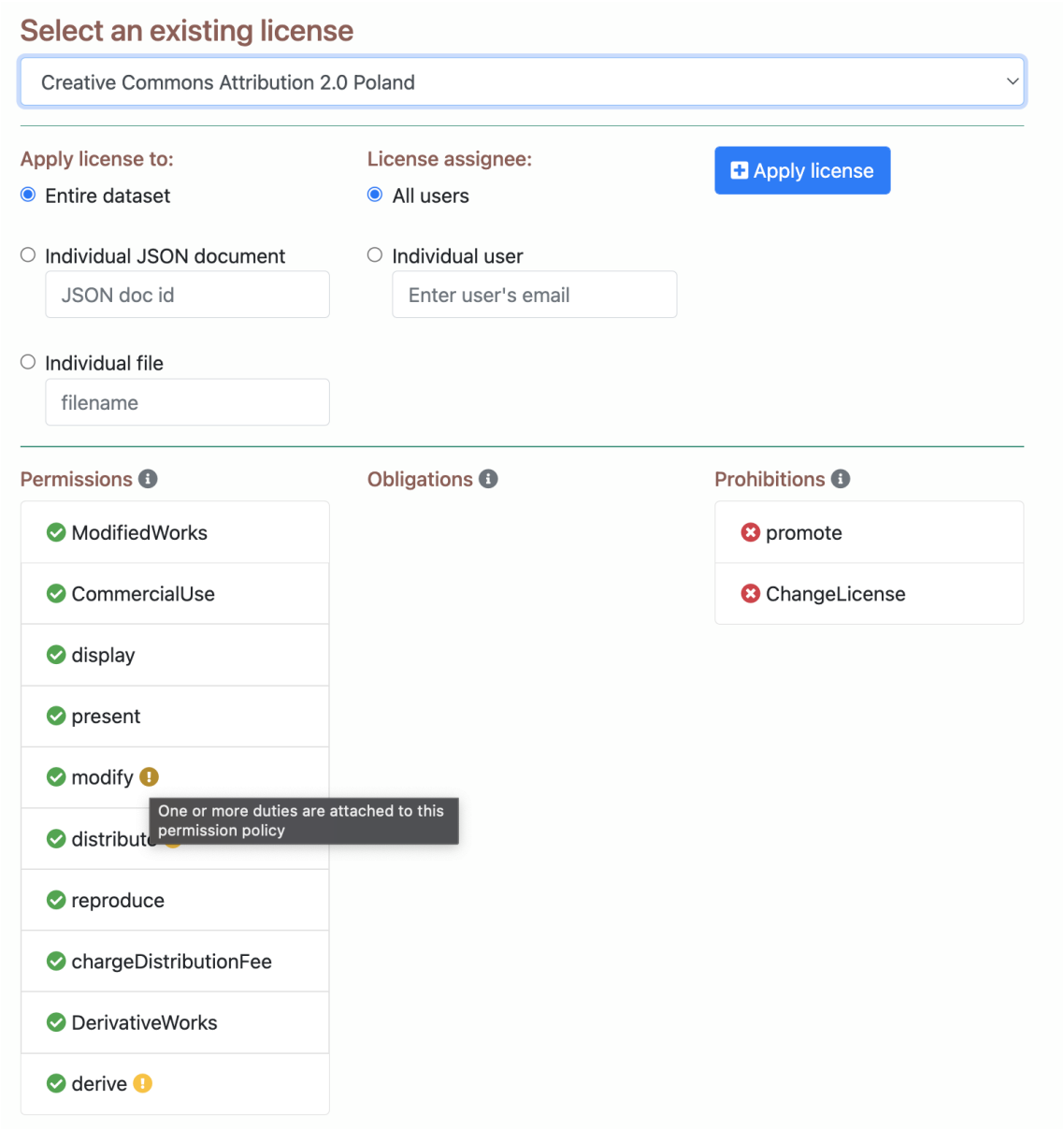

Figure 4.10: Allocating a license to a dataset

4.7 Custom licenses

The standard DALICC-based license library forms the starting point for policy management within the SPICE LDH. Expanding upon this, the policy layer offers dataset owners the ability to generate their own custom licenses. Custom licenses are assembled and stored using the same ODRL-based data format used for managing the standard license library. Starting from either a blank license or using a standard library license as a starting point, dataset owners can add and remove permission, obligation, and prohibition policies using a graphical tool offered in the SPICE LDH portal. This builds on the core licensing data described in section 4.3.2 and further extends the solution to requirements 11 and 12 of this deliverable.

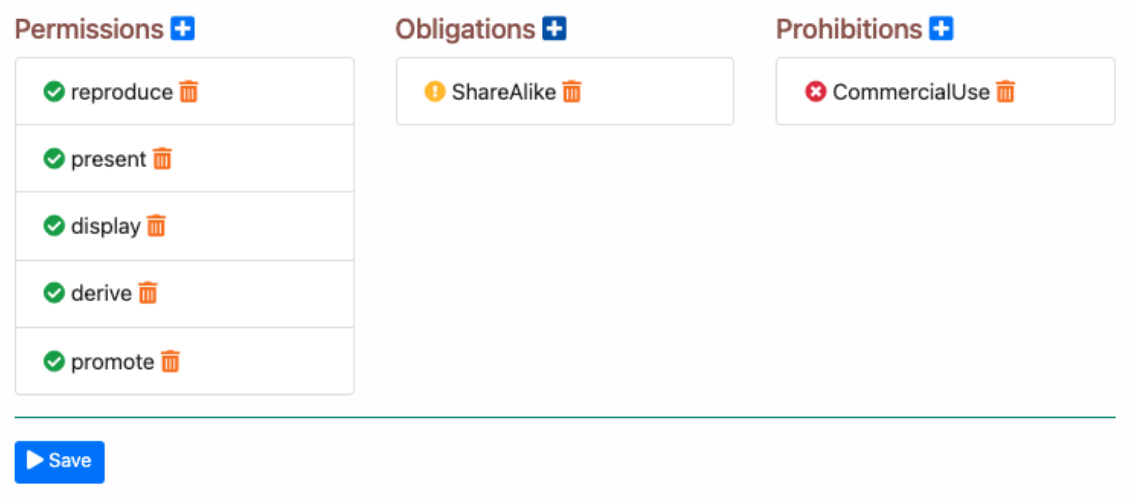

Figure 4.11: Building custom licenses from individual policies

The list of available policies that can be used is derived from the set of all policies that appear across the entire DALICC standard license library. These custom licenses are then stored for future use by dataset owners and managers within the context and scope of the dataset for which they were created. Custom licenses are not transferable across multiple datasets.

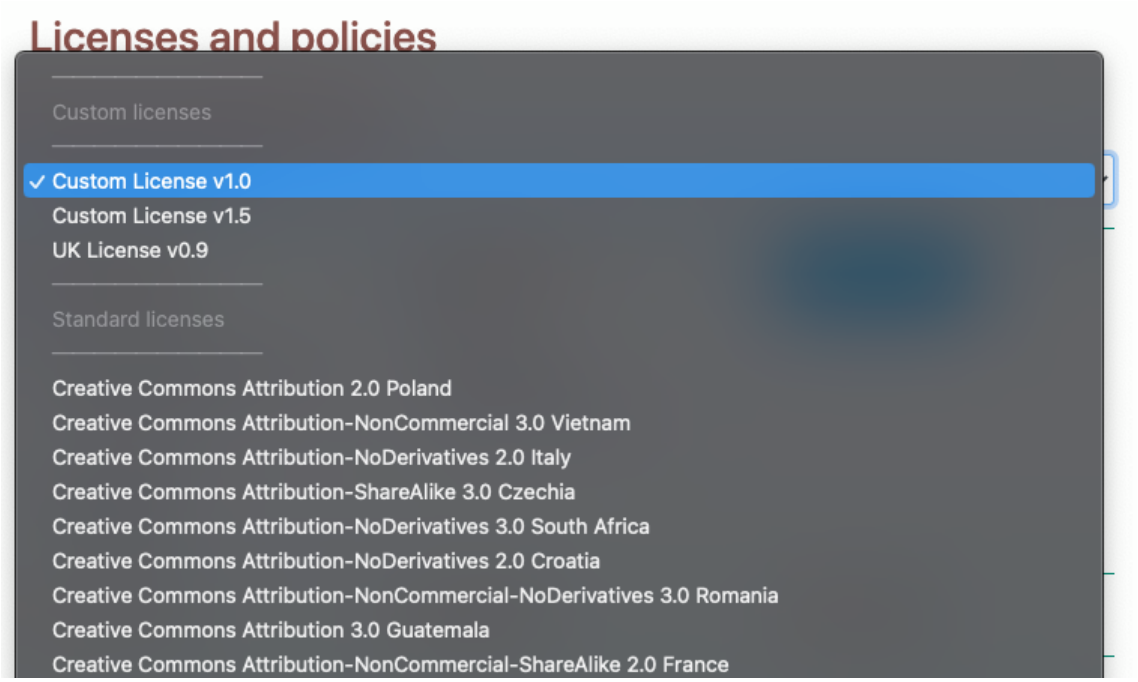

Figure 4.12: Selecting from a list of custom licenses

4.8 Assigning licenses at key registration

All of the work described so far is concerned with allocating policies and licenses to datasets and dataset resources in such a way that they are made available to dataset users. In many cases, only a single license may have been allocated to a dataset for all users and all resources and so all dataset users will be forced to make requests for data under the terms of that single license. However, at the point of a user registering a key for use with a dataset a selection of licenses will be presented (where more than one is available). The user chooses which license is to be used with this particular key registration and that association is stored within the LDH so that the correct license and policies can be served with future LDH API data responses (see section 4.12).

An example of this in use could be that a particular LDH user could be allocated two optional licenses for use on a dataset. That user may create two separate access keys on the dataset, which are passed out to external developers for use in two different applications. Each of these access keys would be paired with a different dataset license, reflecting the agreed terms of use of the particular end-user application. Through this mechanism, application-specific data licensing can be put in place by granular allocation of dataset access keys for individual use cases.

4.9 Resolving license priority

Given the range of license scopes listed above, a dataset may have any number of currently active licenses associated with it. ODRL specification makes no mention of resolving this multi-layered license and policy model for situations where a number of potential licenses must be chosen from before serving digital resources to LDH API users.

The diagram below shows how licenses are layered for a given dataset and the logic that should be applied in order to establish which license takes precedence for a given digital resource request. The licenses taking top priority are any of those that are assigned to both a named user and also a named dataset resource. Following this, licenses applied to individual dataset resources but without a named assignee take the next place in the priority ranking. If no licenses have been applied to the individual dataset resource being requested, any dataset-wide licenses assigned to the current user are applied. Finally, if none of the above cases are matched, the default license for the requested dataset is applied.

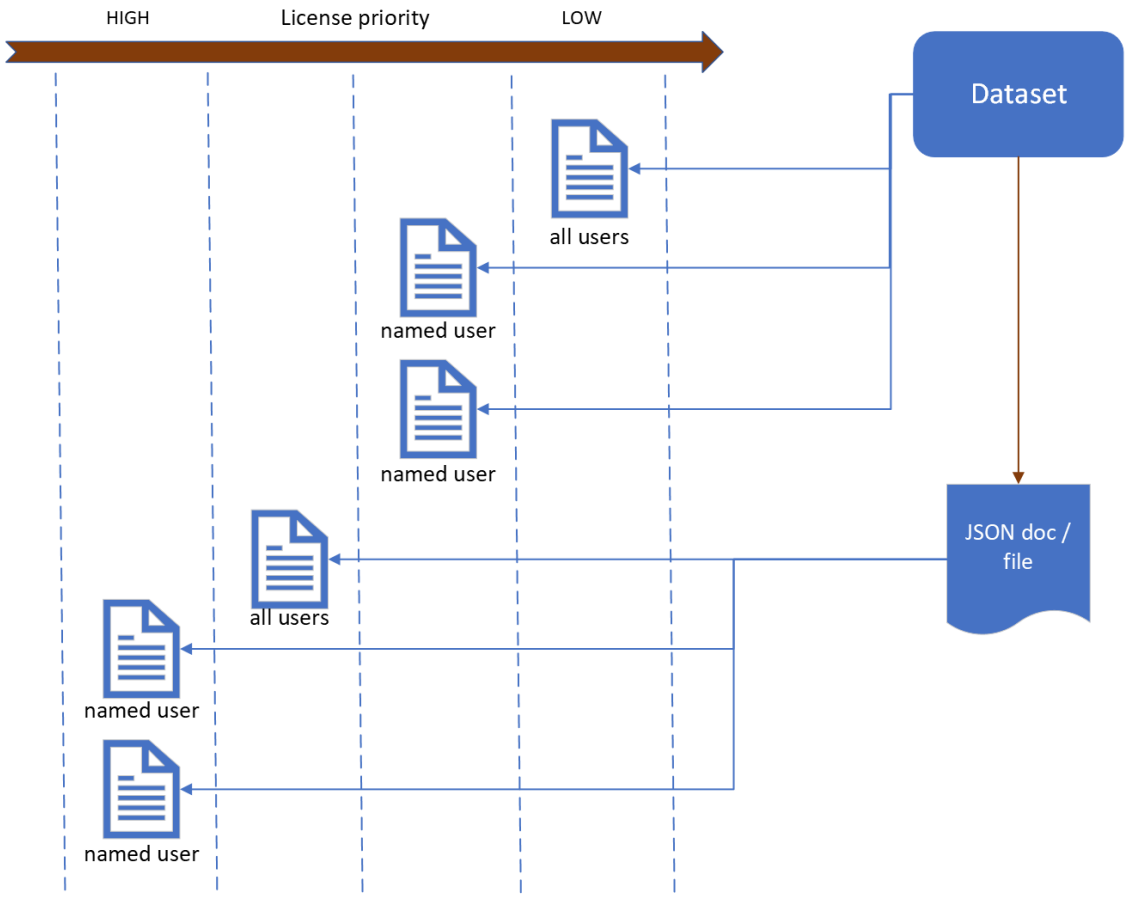

Figure 4.13: Applying the correct license

4.10 License lifespan and historical auditing

Where questions may be raised over disputed use of a dataset or its resources, especially at specific points in the past, it is essential to maintain an audit trail of the license history for each dataset and the resources it contains. The SPICE LDH policy layer makes use of temporal constraint information for each license that has been applied to a dataset. Using this information, the SPICE LDH Portal is able to provide an overview of which licenses were applied at which times, to which sub-resources and to which users. Expressing time constraints (requirement 14) also enables dataset managers to schedule the applicability of licenses, specifying points in time for the activation and deactivation of specific policies should the need arise.

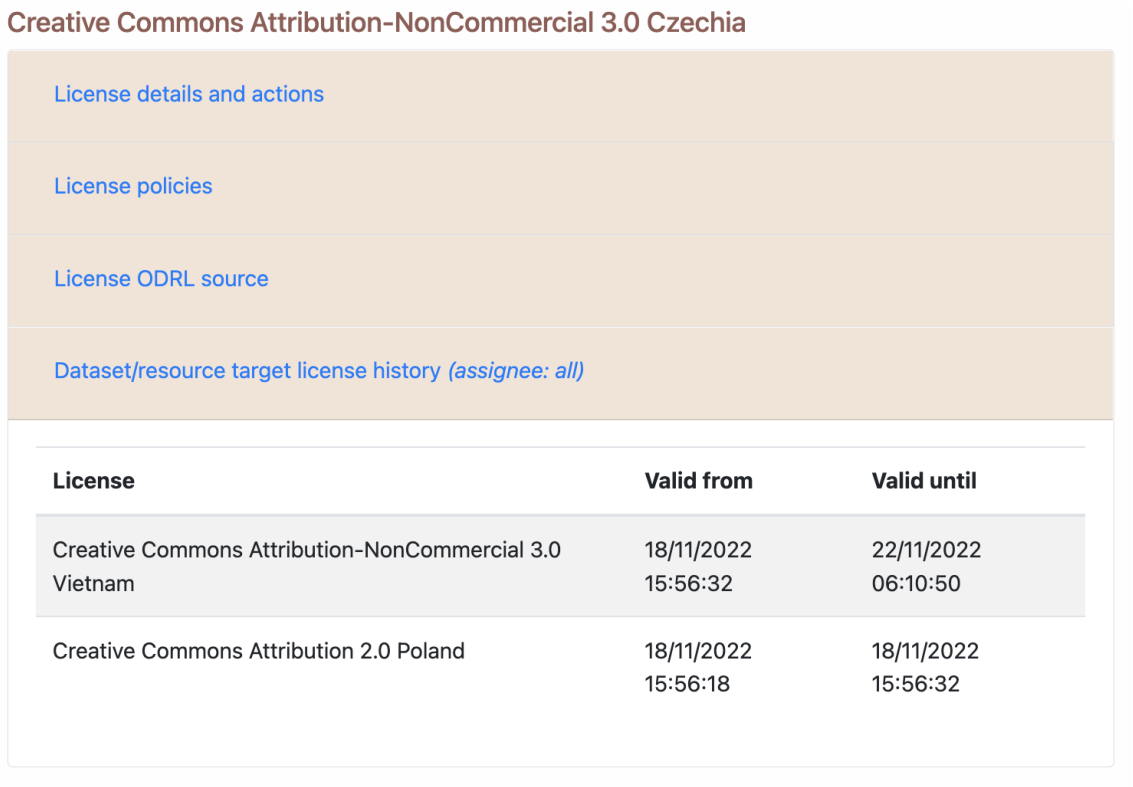

Figure 4.14: Dataset license history

4.11 License and policy negotiation

The current work on the SPICE LDH aims to move beyond the simple allocation of licenses and policies to digital resources and individual users at a granular level. Situations may arise where users' intended use of digital assets stored within the LDH does not conform exactly to the existing set of policies laid out in the current license allocation. In these scenarios, the SPICE LDH aims to offer a process of license and policy negotiation to dataset users whereby they enter into a policy negotiation workflow with the data owner or manager to agree on a specific set of terms to suit their particular use case. This is an iterative process, beginning with an initial request from a user for a specific set of usage terms, followed by a response from the dataset owner (or dataset manager that has been granted the rights to operate as an agent in this workflow, as specified in requirement 26) that may either lead to the granting or rejecting of the request or further counter offers of amended terms that can then enter into a cyclical workflow. The work in this section of the policy management layer specifically addresses requirements 25, 35, and 41, as set out in chapter 3.

During the phase of the workflow where either the dataset user is making a request or the dataset owner is making a counteroffer, sets of policies can be proposed by either selecting an existing license from the standard license library or by building a proposed custom license from a selection of policies. The interface for building these custom licenses from policies is the same as that used by dataset owners for building custom licenses outside of the negotiation workflow, as described in section 4.7

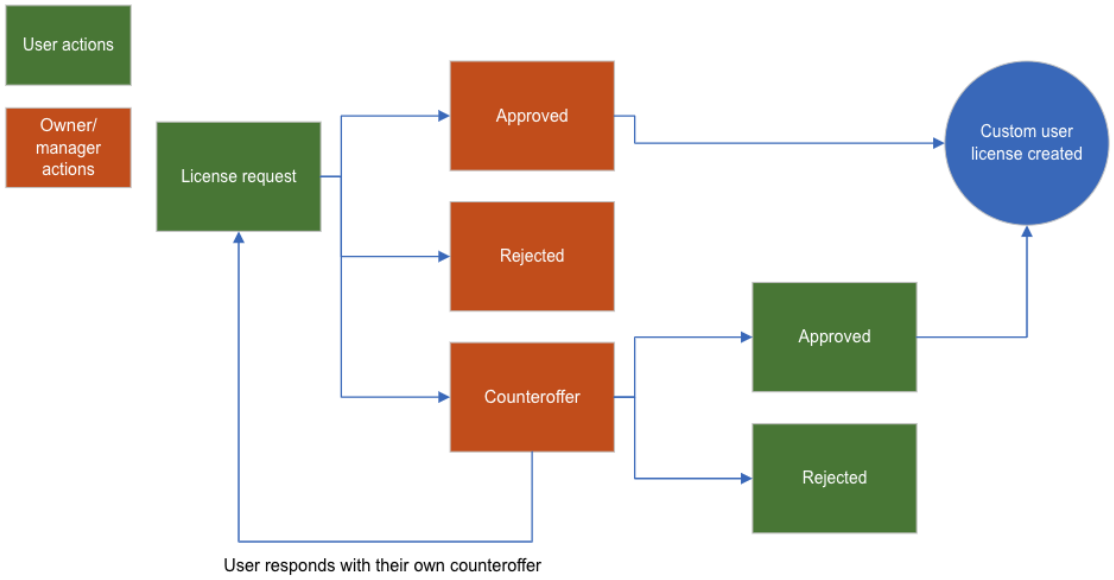

Figure 4.15: Policy negotiation workflow

After a number of iterative steps, outlined above, the status quo is either maintained or the user now has an additional set of policies available to them in the form of a custom license to make use of for the specified dataset or dataset resource. Note that a successful policy negotiation process does not automatically result in a new set of policies being applied to user requests for a specific dataset or resources. The result is simply that a set of policies is now made available to that user for assigning to a chosen access key on the dataset, as described in section 4.8.

4.12 API implementation

The result of all the development, workflows, and structures described above is that licenses can be assigned and served with all LDH API requests for dataset resources. In addition to the licensing and policies module that has been developed for the LDH web portal, the LDH API has been extended to be policy-aware. The API is able to query licensing and policy metadata at the time of data requests being made, perform some reasoning to establish which license applies to a specific request for data, and serve this license information to users along with their data. License information is served with data requests in the form of a custom HTTP response header that only passes a URI reference to the full set of policies. Should the user wish to examine the complete content of the license and associated policies, an additional HTTP API request can be made to a new API endpoint that provides this information in the JSON format described earlier in this report (figure 4.17).

Figure 4.13 shows how multiple licenses can be allocated to a dataset in a stack and how license priority is derived depending on the dataset resource scope and also the LDH user key that is being used to retrieve data. This logic is applied by the API to decide which license URIs to pass to the user in the HTTP response header when data or files are requested.

In cases where a user makes an API request for a single JSON document or file, a single license can be identified and referenced in the response header. However, in cases where multiple items are returned - either a request for the entire contents of the dataset or a query that may deliver an arbitrary number of items - multiple license URIs may be returned to the user. In these cases the API does not have direct knowledge of the list of items being returned, the query results are passed directly from the MongoDB datastore to the API user. Iterating through these results,

often many thousands of documents, and checking license allocation for each one is not feasible without introducing performance issues and unspecified amounts of latency in the API response. For this reason, API requests for multiple items will potentially include multiple license URIs in the response header - the default license that is applied dataset-wide for that user/key and also any resource-specific licenses that are also assigned on that dataset for the user/key making the request.

```
1 Licenses: https://api2.mksmart.org/license/dataset-id/license-id
```

Figure 4.16: Example API response header

Full licenses with policy details can be retrieved from the API, within the context of a dataset, using the URIs provided in the API data response header. As can be seen below a specific license ID can be omitted from the request, this will provide the user with a full list of licenses associated with a given dataset. Note that only the licenses assigned to all users or the specific user making the request will be displayed, since licenses assigned privately to other users may be considered sensitive information.

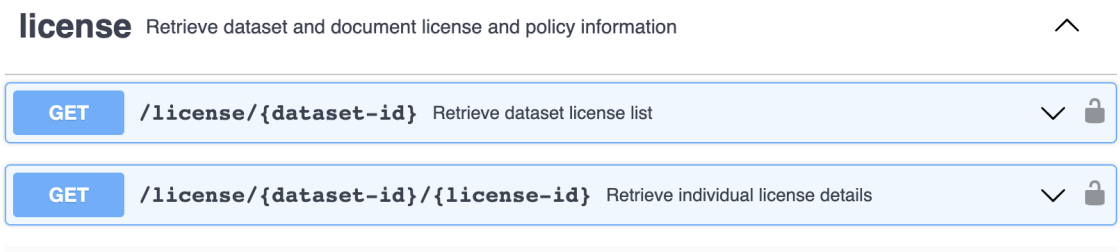

Figure 4.17: API endpoint for license retrieval

4.13 Policy management layer conclusions

In this chapter, we have illustrated how the development of the policy management features of the LDH contributes to the work package and specifically addresses requirements 11, 12, 14, 18, 24, 25, 26, 35, and 41 set out in chapter 3 of this deliverable. Through this work, we offer data owners and managers a sophisticated set of tools for effectively managing the terms of use of their resources. Above and beyond the process of simple allocation of policies, end users are empowered and actively encouraged to be involved in the process. We do this by offering an automated negotiation workflow that lets data users and data managers offer and counteroffer proposed terms of use for specific applications and data resources.

Moving forward, there are areas of the policy management layer that we plan to build upon and develop further. As of this deliverable, the tools developed offer a means of policy expression within the context of dataset resource and user scope. At this stage, the information offered to managers and users is only an expression of terms of use; there aren't any policy reasoning or enforcement components. The system does not make any guarantee that data is being used in accordance with the agreed terms of use, nor does it suggest that curated policies make sense within the context of their other neighbouring policies. For example, there is no mechanism to check that an explicit *permission* that has been applied to a dataset isn't in contradiction with a partnering *prohibition*. Similarly, whilst the system can reason on which of a number of layered licenses to apply in a particular scenario, it is not aware of how some policies are more or less restrictive than others. With more work in this area, it would be possible to inform dataset managers of the implications of applying one license versus another, specifically in terms of whether one license either further restricts or in fact relaxes the terms of use on their data in relation to another license.

5 The Linked Data Hub: privacy monitoring layer

This section of the report details the SPICE privacy monitoring layer developed in this deliverable. The privacy monitoring layer is one of the components of LDH content monitoring. The report outlines the background and purpose of the module, provides its technical requirements, highlights the application workflow, gives the approach and methodology, and presents feedback on how the module integration with the SPICE Linked Data hub (LDH) has developed so far. It concludes by outlaying considerations for moving forward with further development.

5.1 Background and purpose

Large amounts of content generated during citizen curation get fed into the memory of institutions. This phenomenon is problematic because data managers cannot control the content of their data. It implies that the data provided may not comply with the data management policies, for example, GDPR, which is the data protection framework of the European Union (EU). This means that the data managers are responsible for monitoring the composition of these contents to know whether any personally identifiable information (PII) has been recorded. For this purpose, we developed a content monitoring solution that automatically flags personally identifiable information (PII), helping data managers comply with the General Data Protection Regulation (GDPR). At present, there may be potential privacy violations in LDH datasets because there is no monitoring of the user-generated content from citizen curation activities. The privacy monitoring module is being developed and focused on identifying and alerting data managers of potential privacy breaches within LDH content.

5.2 Requirement specification

This subsection describes what the privacy monitoring solution will do and how it will be expected to perform. Two solution requirements were considered during the development of the privacy monitoring solution. These include functional and non-functional requirements.

The functional requirements present the features and functions of the privacy monitoring solution. These requirements include the following:

- The data managers should be alerted when PII is found.
- The data managers should be notified of the document ID within their dataset where the PII is found.
- The data managers should be notified of the fields within their document in which PII alerts were generated.
- The data manager should be notified of the types of PII found.
- The data managers should be notified of a severity score of the PII found.
- The data managers should be able to take action on these notifications.

The non-functional requirements provide the quality attributes of the privacy monitoring solution. These include:

- The system should be able to handle all citizen-curated activity without performance deterioration.
- The system should be able to be integrated into the current SPICE-linked Data Hub as one of the LDH content monitoring modules (figure 5.1).
- The system should use an application programming interface (API) to interact with the LDH activity log.
- The system should be able to flag appropriate PII notifications to data managers
- The system should be able to scan all updates and changes on a datasets for privacy breaches.

- The system should be able to run every minute.
- The system should be able to keep track of its progress through the use of a timestamp that indicates which LDH data changes have been scanned so far.

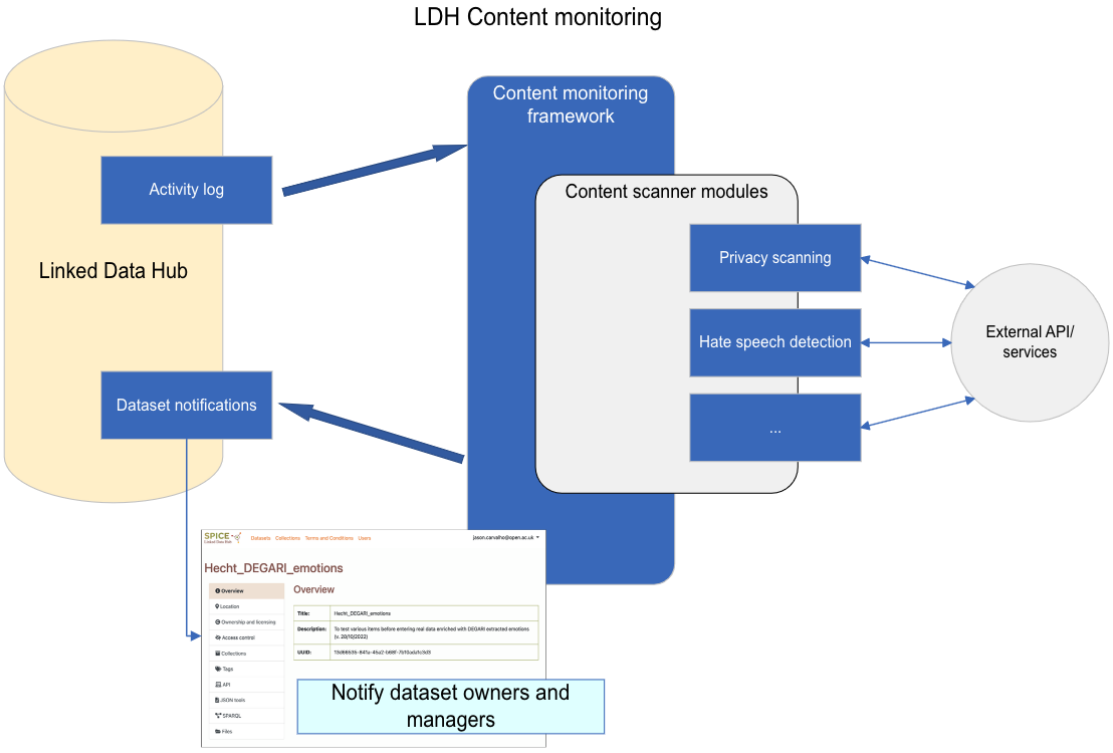

Figure 5.1: LDH content monitoring

5.3 Application workflow

In this subsection, the processes and modules, as well as the implementation procedure required in developing the privacy monitoring solution, are presented.

The application workflow of how the privacy monitoring solution checks and notifies the data managers of privacy violations is presented in figure 5.2. As citizen curation activities occur at the LDH, the privacy monitoring solution calls the LDH activity log via an API. On getting a response, the monitoring solution extracts the content of the “payload” from the JSON documents and iterates over it, field by field, to identify potential PII. If PII is found, the system alerts the dataset owners through a dataset’s notifications tab in the LDH portal.

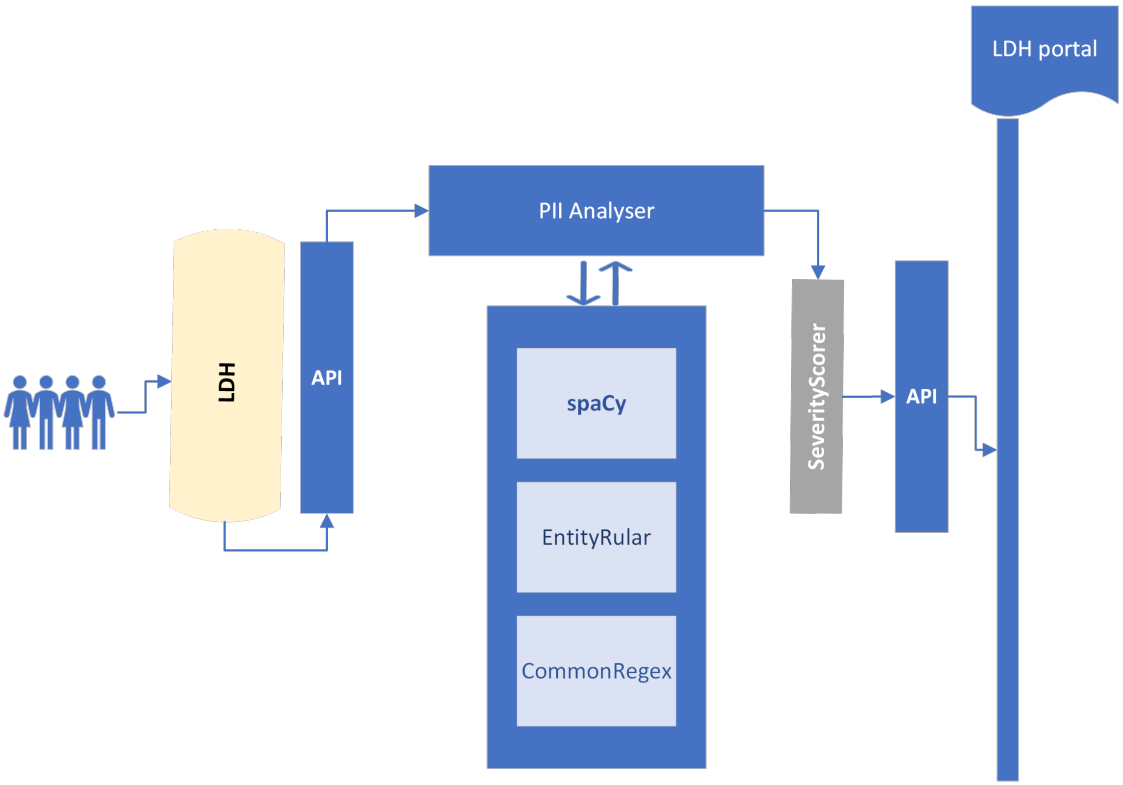

Figure 5.2: Privacy monitoring solution application workflow

5.4 Approach to detecting PII

This section presents the approaches we adopted in detecting PII in user-generated content. It details the tools used, the severity scores used, and the algorithmic description.

5.4.1 Tools

In order to monitor personally identifiable information effectively, we made use of state-of-the-art tools and libraries for natural language processing. These tools include spaCy, EntityRuler, and Common regular expression. The spaCy library is an open-source python library for advanced natural language processing [30]. EntityRuler is a spaCy factory that allows one to create a set of patterns with corresponding labels [31]. Common regular expression is an open-source python library to detect dates, times, emails, phone numbers, links, emails, IP addresses, prices, emails, street addresses, and more within a string. [32]. The combination of these tools gave rise to a robust and high computational system capable of detecting a total of fourteen (14) types of PII, as presented in table 5.1.

5.4.2 Severity scores

In order to standardise the severity of PII found for data managers, we designed a severity model. Table 5.1 presents a list of PII and their severity scores. Here is the method we used to compute the severity score. We classify the PII according to how directly they identify a person or entity. These include direct identifiers, indirect identifiers, mild identifiers, and non-identifiers.

- All direct identifiers, such as a person's name, phone number, credit card details, social media handles, and email addresses, are classified as "Extreme" with a score of 4.
- All indirect identifiers, such as street name and IP address, are classified as "High" with a score of 3.
- All mild identifiers, such as organisations, links, and postcodes, are classified as "Medium" with a score of 2.
- All non-identifiers to person, such as location, age, date, and time, are classified as "Low" with a score of 1.

The overall severity score is determined based on the sum of all the points from the severity factors found in a document.

Table 5.1: List of PII and severity score.

PII	**Severity**	**PII**	**Severity**
Person	Extreme	Date	Low
Location	Low	Time	Low
Organisation	Medium	Postcode	Medium
PhoneNumber	Extreme	Email	Extreme
Age	Low	Street name	High
Links	Medium	SocialMedia	Extreme
CreditCard	Extreme	IPS	High

5.4.3 Algorithmic description of the severity model and algorithm

To ensure that the privacy monitoring solution performs well in the severity scoring, we developed an algorithm for the severity score (figure 5.3). Once user-generated content is scanned for a privacy violation, the results are fed into the severity function for scoring. We assigned numbers to all the PII based on their classification as described in the severity score subsection. So, if just one PII is found, for example, age, the system reports an individual score of that PII as the severity score. Alternatively, if more than one PII is found after scanning user-generated content, the system calculates the severity score by adding the scores of individual entities. We benchmarked the severity scores at 4, meaning that a combined computation equal to 4 and above will be regarded as 4, which is Extreme. For example, when age (score=1) and street name (score=3) are detected, the system adds these scores to rank the severity score. This new severity score of Extreme implies that age and street name have become direct identifiers which they may not be on an individual basis.

Below are some of the scoring models.

- Any 2 High = Extreme
- Any 2 medium = Extreme
- Any 1 high + 1 medium = Extreme
- Any 1 medium + 1 low = High
- Any 2 low + 1 medium = Extreme
- Any 1 high + 1 low = Extreme
- Any 4 low + Extreme
- Any 2 low = Medium

```
def severityScores(*value):
        checkLow = locationEntity + dateEntityd + \
                   timeEntityd + ageEntityd + streetNameEntityd
        checkExtreme1 = socialMediaHandlesEntity + \
                   streetNameEntity  # Any 2 High = Extreme
        checkExtreme2 = organisationEntity + postcodeEntityd + \
                   linksEntityd + IPSEntityd  # Any 2 medium = Extreme
        checkExtreme3 = checkExtreme1 + checkExtreme2  # Any 1 high + 1 medium = Extreme
        checkHigh = checkExtreme2 + checkLow  # Any 1 medium + 1 low = High
        checkExtreme4 = checkHigh + checkLow  # Any 1 high + 1 low = Extreme
        checkExtreme5 = checkLow + checkExtreme2  # Any 2 low + 1 medium = Extreme
        checkExtreme6 = checkLow  # 3 low + Extreme
        checkMedium = checkLow  # Any 2 low = Medium

        severityScores = ''
        colorCode = ''
        if nameEntity > 0 or emailEntityd > 0 or phoneNumberEntity > 0 or creditEntityd > 0:
            severityScores = 'extreme'
            colorCode = '#F00'
        elif checkExtreme1 > 0 or checkExtreme2 > 0 or checkExtreme3 > 0 or checkExtreme4 > 0
        or checkExtreme5 > 0 or checkExtreme6 > 2:
            severityScores = 'extreme'
            colorCode = '#F00'
        elif checkHigh > 0:
            severityScores = 'high'
            colorCode = '#FFC0CB'
        elif checkMedium == 2:
            severityScores = 'medium'
            colorCode = '#FFFF00'

        return {
            "severityScores": severityScores,
            "colorCode": colorCode
```

Figure 5.3: Algorithm for Severity Scores

5.4.4 Approach, Data, and Testing

Developing an approach to ascertain the privacy monitoring solution's effectiveness in detecting and identifying PII was important. The strategy taken was divided into two parts: the first with generic data and the second with publicly available contracts having PII. These strategies become necessary as we currently are not aware of any existing LDH datasets that have suitably reliable numbers of instances of PII for our development.

Firstly, we gathered datasets that had previously been annotated. We used Kaggle, a website community for sharing and enhancing datasets, for this aim. The datasets used in the testing were built from the Groningen Meaning Bank [33]. Secondly, it was necessary to use the PyPDF2 Python module to extract the data from the publicly available contract found in PDF format containing PII and transform it into text files. After that, it was manually tagged. We used the annotated datasets and manually tagged datasets as gold standards to determine the extent to which the monitoring solution can accurately identify PII.

For the system to be tested, we stored these datasets in the LDH as a dataset. Then, allow the monitoring solution to call the text from the API and analyse the user-generated content. Our testing has shown encouraging results. We plan for further accuracy testing on a wider range of data and will report this on the next deliverable.

5.4.5 Method for alerting data managers

As the privacy monitoring solution runs every minute, it sends notifications to the respective dataset owners via the LDH portal. Once the alert is sent to the LDH portal, it appears in the notification tab where the dataset owner can click, view, and take appropriate action on the flagged PII. The PII notifications are presented amongst other notifications that may have been generated by other content monitoring modules. These notifications are filterable by type, giving users the ability to just view privacy-related messages, and include all the information about the suspected privacy violation that was recorded at the time of scanning. The source data for a typical notification is shown in figure 5.4.

```
notification = {
          "job-type": "PRIVACY-VIOLATION",
          "submitted-by": "LDH-SCANNER",
          "modified": 1671148722,
          "message": "Personally Identifiable information was detected in this
             doscument",
          "severityScores": "Extreme",
          "flags": [
                {
              "key": "artwork_thoughts",
              "value": "This reminds me of where I live in Paris",
              "pii": {
                  "type": "location",
                  "description": "A geographical entity was detected",
                 }
           "alertScore": "low"
        }

        {
          "key": "other_comments",
          "value": "Please contact me at david@hotmail.com for more information"
             ,
          "pii": {
              "type": "Email"
              "description": "An email address was detected"
           }
          "alertScore": "Medium"
        }
      ]
          "dataset": "pii_testing",
          "document": "60c0908cb5b26479a17d1b33",
          "datasetid": "spice_rdfjobs2",
          "status": "ALERT",
        }
```

Figure 5.4: Summary of flagged PII notification

The notification tab located at the left-hand side of the LDH portal is shown in figure 5.5. A drop-down menu allows the dataset managers to filter the notification types or select all notifications from the LDH content monitoring.

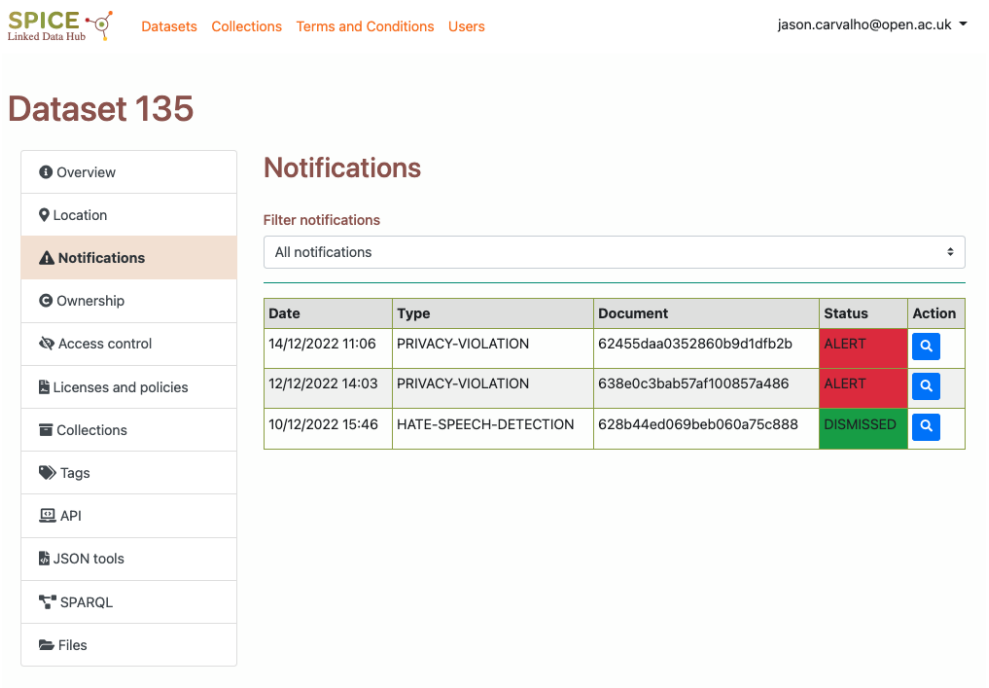

Figure 5.5: Dataset notifications within the Linked Data Hub

Figure 5.6 shows the expanded view of a selected PII notification, where the user is able to examine the overall PII severity for this document. It also gives a breakdown of the individual fields that were flagged in the document, the PII type that was identified for each field, and the corresponding severity score for each flagged field.

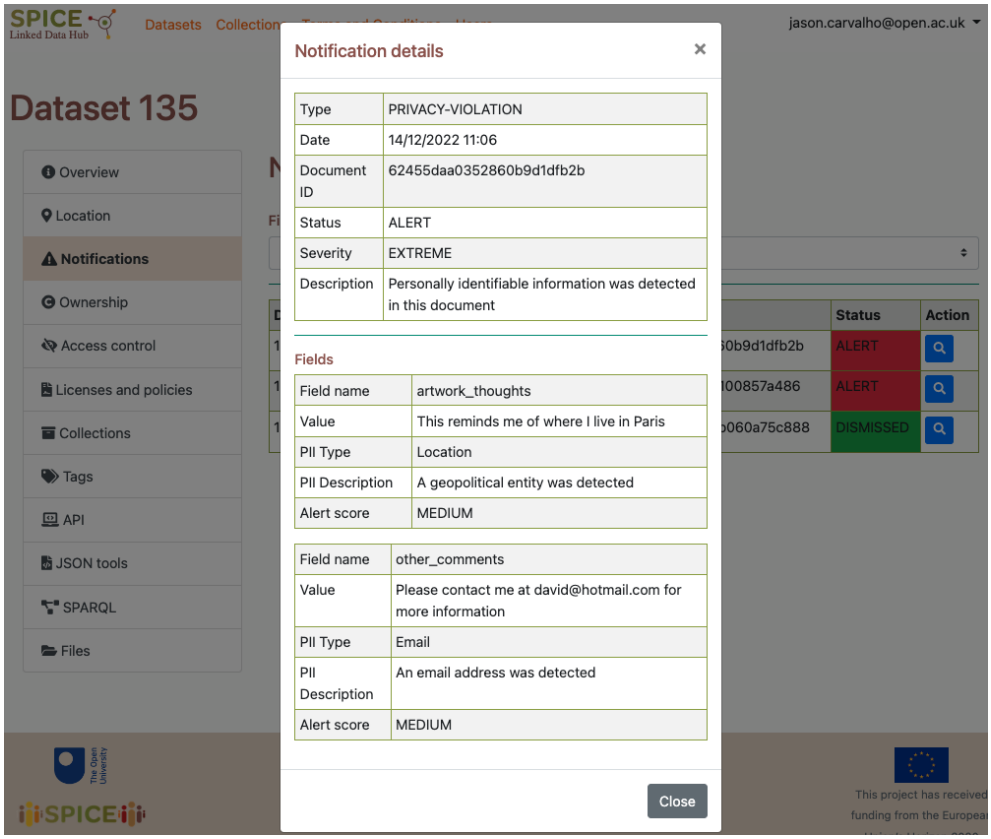

Figure 5.6: Expanded detail view for a selected notification

5.5 Privacy monitoring layer conclusions

LDH infrastructure and API have proven to be notably dependable and responsive throughout the development and testing of the privacy monitoring solution. The monitoring solution, for the moment, is designed to analyse content in the English language. This means that privacy violations in other languages will not be flagged. This gap will be resolved by incorporating a multi-lingual model into the system. It is believed that LDH infrastructure can accommodate this lingual model. However, as the project nears closure, the appropriate time for its implementation remains challenging. We hope to include the multi-lingual model in the next version of the privacy monitoring solution.

6 Conclusions

In this deliverable, we presented the Privacy and Policy layer of the SPICE Linked Data Hub. We reviewed relevant literature on metadata management systems, focusing on methods for representing and managing terms of use and data policies, and highlighted privacy issues deriving from incorporating user-generated content. We then reported on the progress documented by this deliverable in relation to the requirements of the SPICE Linked Data Hub, originally introduced in D4.1 [1]. In Chapter 4 we described the new functionalities of the policy management layer, from the ability to design custom terms and conditions, to the methods for assigning them to applications or negotiating them between data managers and users. Finally, we described in Chapter 5 a content monitoring system that was developed for the Linked Data Hub and its application to the case of detecting privacy violations, for supporting data managers in assessing the GDPR compliance of the collected user-generated content. In the future, we aim at expanding the content monitoring functionality of the LDH by incorporating other intelligent content analysers, for example, the hate speech detection system developed in the context of WP3. Finally, the work presented in this deliverable will be complemented by *D4.6 Provenance and Process analysis layer: Supporting use cases*, where we will detail the API functionalities also related to policy assessment as part of the provenance information, and *D4.7 Linked Data server technology: Final release and open-source distribution*, where we will finalise the software for distribution and use beyond the scope of the SPICE case studies.

Bibliography

[1] E. Daga, P. Mulholland, J. Carvalho, R. Damiano, M. Daquino, B. D. Agudo, T. Kuflik, A. Lieto, and A. Bosca, "Linked data server technology: requirements and initial prototype," *SPICE Project, European Union's Horizon 2020 grant agreement No 870811*, vol. Deliverable 4.1, 2021.

[2] E. E. Daga, "Linked data server technology: integrating feedback from use case requirements," *SPICE Project, European Union's Horizon 2020 grant agreement No 870811*, vol. Deliverable 4.2, 2022.

[3] "Stakeholders' Survey on a European Collaborative Cloud for Cultural Heritage. Report on the online survey results," European Commission Directorate-General for Research and Innovation Directorate D — People Unit D.3 — Fair Societies & Cultural Heritage, 2022.

[4] D. McCarthy and A. Wallace, "Survey of glam open access policy and practice," *Copyright Cortex.*, vol. 18, p. 2020, 2018. [Online]. Available: http://bit.ly/OpenGLAMsurvey

[5] D. Scott and R. Sharp, "Abstracting application-level web security," in *Proceedings of the 11th international conference on World Wide Web*, ACM. ACM, 2002, pp. 396–407.

[6] R. Iannella, "Open digital rights management," in *World Wide Web Consortium (W3C) DRM Workshop*. Sophia-Antipolis, France: W3C, 2001.

[7] V. R. Benjamins, P. Casanovas, J. Breuker, and A. Gangemi, "Law and the semantic web, an introduction," in *Law and the Semantic Web*. Springer, 2005, pp. 1–17.

[8] V. Borissova, "Cultural heritage digitization and related intellectual property issues," *Journal of Cultural Heritage*, vol. 34, pp. 145–150, 2018.

[9] H. Stitzlein, M.-J. K. Han, and S. R. Benson, "Unraveling challenges: Rights statements in digital cultural heritage collections," *Journal of Library Metadata*, vol. 18, no. 3-4, pp. 135–150, 2018.

[10] P. A. Bonatti and D. Olmedilla, "Rule-based Policy Representation and Reasoning for the Semantic Web," in *Proceedings of the Third International Summer School Conference on Reasoning Web*, ser. RW'07. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 240–268. [Online]. Available: http://dl.acm.org/citation.cfm?id=2391482.2391488

[11] S. Kirrane, S. Villata, and M. d'Aquin, "Privacy, security and policies: A review of problems and solutions with semantic web technologies," *Semantic Web*, vol. 9, no. 2, pp. 153–161, 2018.

[12] S. Villata and F. Gandon, "Licenses compatibility and composition in the Web of data," in *Proceedings of the Third International Conference on Consuming Linked Data-Volume 905*, CEUR-WS.org. CEUR-WS, 2012, pp. 124–135.

[13] G. Governatori, A. Rotolo, S. Villata, and F. Gandon, "One license to compose them all," in *The Semantic Web–ISWC 2013*. Sydney, Australia: Springer, 2013, pp. 151–166.

[14] E. Daga, A. Gangemi, and E. Motta, "Reasoning with data flows and policy propagation rules," *Semantic Web*, vol. 9, no. 2, pp. 163–183, 2018.

[15] R. Iannella, M. Steidl, S. Myles, and V. Rodríguez-Doncel, "ODRL Version 2.2 Ontology," W3C, Tech. Rep., 2017. [Online]. Available: https://www.w3.org/ns/odrl/2/

[16] R. Pucella and V. Weissman, "A formal foundation for odrl," *arXiv preprint cs/0601085*, 2006.

[17] V. Rodriguez-Doncel, S. Villata, and A. Gómez-Pérez, "A dataset of rdf licenses." in *JURIX*, 2014, pp. 187–188.

[18] C. Cardellino, S. Villata, F. Gandon, G. Governatori, B. Lam, and A. Rotolo, "Licentia: a Tool for Supporting Users in Data Licensing on the Web of Data," in *Proceedings of the ISWC 2014 Posters & Demonstrations Track, a track within the 13th International Semantic Web Conference (ISWC 2014)*, M. Horridge, M. Rospocher, and J. van Ossenbruggen, Eds., Riva del Garda, Italy, 21 October 2014.

[19] E. Daga, M. d'Aquin, E. Motta, and A. Gangemi, "A Bottom-Up Approach for Licences Classification and Selection," in *The Semantic Web: ESWC 2015 Satellite Events*, Springer. Portorož, Slovenia: Springer, 2015. [Online]. Available: https://link.springer.com/book/10.1007%2F978-3-319-25639-9

[20] H.-P. Lam and G. Governatori, "The making of spindle," in *International Workshop on Rules and Rule Markup Languages for the Semantic Web*, Springer. Las Vegas, Nevada (USA): Springer, 2009, pp. 315–322.

[21] T. Pellegrini, V. Mireles, S. Steyskal, O. Panasiuk, A. Fensel, and S. Kirrane, "Automated rights clearance using semantic web technologies: The dalicc framework," in *Semantic Applications*. Springer, 2018, pp. 203–218.

[22] A. Oltramari, D. Piraviperumal, F. Schaub, S. Wilson, S. Cherivirala, T. B. Norton, N. C. Russell, P. Story, J. Reidenberg, and N. Sadeh, "PrivOnto: A semantic framework for the analysis of privacy policies," *Semantic Web*, vol. 9, no. 2, pp. 185–203, 2018.

[23] P. A. Bonatti, S. Kirrane, I. M. Petrova, and L. Sauro, "Machine understandable policies and gdpr compliance checking," *arXiv preprint arXiv:2001.08930*, 2020.

[24] I. R. S. W. Group *et al.*, "Rightsstatements. org white paper: Recommendations for standardized international rights statements," *Retrieved from rightsstatements. org/files/160208recommendations_for_ standardized_international_rights_statements_v1*, vol. 1, 2015.

[25] S. R. Benson and H. Stitzlein, "Copyright and digital collections: A data-driven roadmap for rights statement success," *College & Research Libraries*, vol. 81, no. 5, p. 753, 2020.

[26] K. N. Vavliakis, G. T. Karagiannis, and P. A. Mitkas, "Semantic web in cultural heritage after 2020," in *Proceedings of the 11th International Semantic Web Conference (ISWC)*, 2012, pp. 11–15.

[27] L. McKenna, C. Debruyne, and D. O'Sullivan, "Understanding the position of information professionals with regards to linked data: a survey of libraries, archives and museums," in *Proceedings of the 18th ACM/IEEE on Joint Conference on Digital Libraries*. ACM/IEEE, 2018, pp. 7–16.

[28] E. Daga, L. Asprino, R. Damiano, M. Daquino, B. D. Agudo, A. Gangemi, T. Kuflik, A. Lieto, M. Maguire, A. M. Marras *et al.*, "Integrating citizen experiences in cultural heritage archives: requirements, state of the art, and challenges," *ACM Journal on Computing and Cultural Heritage (JOCCH)*, vol. 15, no. 1, pp. 1–35, 2022.

[29] E. Daga, M. d'Aquin, A. Adamou, and E. Motta, "Addressing exploitability of smart city data," in *2016 IEEE International Smart Cities Conference (ISC2)*. Trento, Italy: IEEE, 2016, pp. 1–6.

[30] Y. Vasiliev, *Natural Language Processing with Python and SpaCy: A Practical Introduction*. No Starch Press, 2020.

[31] N. K. Sivaraman, R. Koduri, and M. Manalikandy, "A hybrid method for automotive entity recognition," SAE Technical Paper, Tech. Rep., 2021.

[32] H. Yang, H. Lu, S. Li, M. Li, and Y. Sun, "Research on content extraction of rich text web pages," in *International Conference on Artificial Intelligence and Security*. Springer, 2019, pp. 279–287.

[33] J. Bos, V. Basile, K. Evang, N. J. Venhuizen, and J. Bjerva, "The groningen meaning bank," in *Handbook of linguistic annotation*. Springer, 2017, pp. 463–496.