decode

Legal Ontology to Support Smart Contracts in DECODE Scenarios.

Ontology Methodology, Building Blocks, Examples

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no. 732546

Project no. 732546

DECODE

DEcentralised Citizens Owned Data Ecosystem

D3.5-Appendix Legal Ontology to Support Smart Contracts in DECODE Scenarios. Ontology Methodology, Building Blocks, Examples

Version Number: V1.0

Lead beneficiary: UOC

Due Date: [Month Year]

Author(s): Monica Palmirani, University of Bologna, CIRSFID

Editors and reviewers: Andrea D'Intino (DYNE.ORG), Marco Ciurcina (NEXA), Antonio Calleja (UOC)

Dissemination level:		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Approved by: Francesca Bria (Chief Technology and Digital Innovation Officer, Barcelona City Hall)

Date: 31/12/2019

This report is currently awaiting approval from the EC and cannot be not considered to be a final version.

Contents

1. Objective

This document aims to present the methodology for building a DECODE ontology useful to support the DECODE goals, especially in relation to smart contracts i) for checking compliance in the personal data protection domain; ii) for managing the data sovereignty; iii) for the creation of data commons. A little proof-of-concept is presented in this deliverable. It is not an objective of this work to build a DECODE ontology that needs, by definition, at least one year of analysis, modelling, representation, evaluation, validation and refinement. The main aim is to provide a methodology for building DECODE ontology, to evaluate pros and cons concerning technical solutions, to provide some examples integrated with the scenario of DECODE.

2. Ontology for DECODE

Including ontology in DECODE project produces the following benefits:

1. improve the creation of a common vocabulary in the community focused on a given domain. Especially in the public administration.
2. support the NLP layer for correctly detecting the labels in the text useful for the rule modelling;
3. produce a preliminary checking compliance of the rules using description logic reasoning;
4. provide a good tool for enhancing the query the data;
5. connect the output of the smart contract with the ontology concept and so to provide more information;
6. make more explainable the smart contract according to the Guide Lines of the High-Level Expert Group on Artificial Intelligence;
7. use ontology for modifying unexpected situations using oracles (e.g., modification of a norm parameter).

In general, it is fundamental in the legal domain to adopt legal ontology for classifying concepts (VAN KRALINGEN 1997, HOEKSTRA), to define functions of the legal system using relationships (VALENTE 1994), for capture the judiciary system case-law reasoning (CASELLAS 2011, Roussey 2011). In general, it is fundamental to have ontology in legal domain in order to modelling the different kind of legal data that can intervein in the smart contract. See the picture below that attempts a first taxonomy of the different kind of legal data by functions.

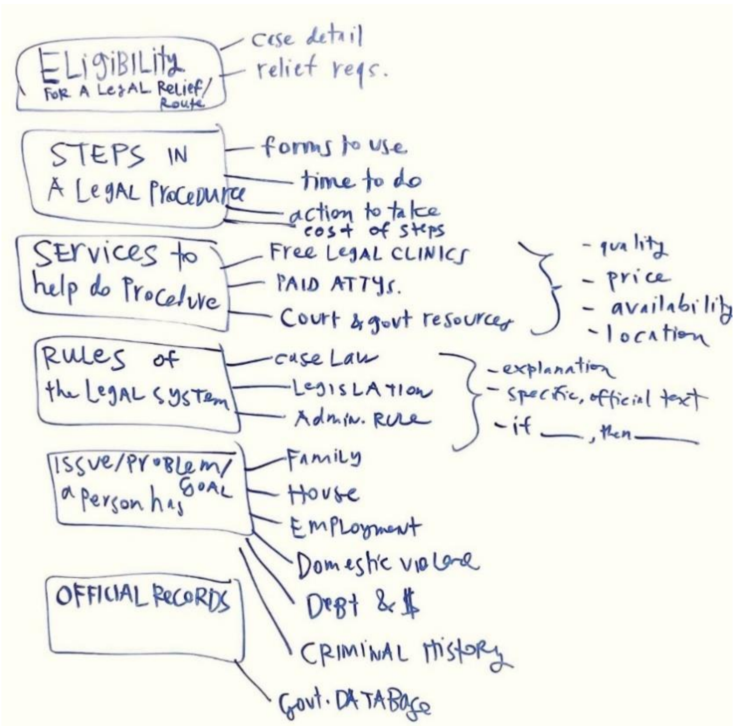


Figure 1 - An example ontology for categorizing legal data; Source: Margaret Hagan

3. Scenarios to Address

3.1 Petition

Situation: We are in 2025. The Municipality of Orion set up a digital petition concerning the geographic area where to build the new airport. Only the adults, citizenships and living in proximity of the selected geo-area are eligible to vote the petition. Using *zenroom* technologies the application asks to verify the following pre-requirements in the credential layer:

- Physical person with the right to vote in a given time;
- Citizen of Orion in a given time;
- Living in given area in a given time (neighbourhood).

The petition is valid if the voting people reaches the quorum of 2/3 of people with the right to vote according to the regulation of the Municipality Orion. If the petition is valid then we must count the votes. The petition is passed using the simple majority. The result could be positive, negative.

In this narrative of the use-case we can identify several actors, definitions, conditions that can affect a smart contract:

- What is the definition of physical person?
- What is the definition of right to vote?
- How to check the correct geographic conditions?
- How to manage the concept of quorum when it evolves over the time?
- How to manage the results of the petition?

Those questions depend to the legal system and to the domain.

If those parameters are part intrinsically part of a smart contract (also intending a set of smart contract each other connected in pipeline) how to manage the evolution of those concepts over the time, space, jurisdiction, legal system?

The main idea is to use ontology connected with an abstract framework of smart contracts where parameters come from RDF assertions, managed by a knowledge graph. In the following case the general rules included in the smart contract could be integrated by the ontology that can customize the general rules with the specific rules coming from the jurisdiction. In case, of voting rules we can includes specific requirements according to the situation and the member state. E.g. If we are voting for the school services, also the minors could vote but only if the rights that permits this expression of vote are sufficiently strong.

For this reason, a query could be made to the legal ontology for collecting customized further rules according to the situation and the eGov service, like in the example below.

General Smart contract rules for voting	**Customization after the query to the legal ontology concerning the specific requirements about the given situation and jurisdiction**

`PhysicalPerson(?p) ∧`
`hasAge(?p, ?a) ∧`
`hasRightToVote(?p,?r) ∧`
	`PhysicalPerson(?p) ∧`
	`hasAge(?p,?a) ∧`
	`hasRightToVote(?p,?r) ∧`

City(?c) ∧ hasCitizen(?c, ?p,) ∧ Neighborhood (?n) ∧ hasInhabitant(?n, ?p,) ∧ swrlb:greaterThanOrEqual(?a, ?ar) → canVote(?p, ?n)	City(?c) ∧ hasCitizen(?c, ?p) ∧ Neighborhood (?n) ∧ hasInhabitant(?n, ?p) ∧ **Situation(?s) ∧** **hasAgeReq (?s, ?ar) ∧** **hasJurisdiction(?s, ?j) ∧** **hasRightReq(?s, ?rq) ∧** swrlb:greaterThanOrEqual(?a, ?ar) swrlb:matches(?r, ?rq) → canVote(?p, ?s)

3.2 Smart city IoT data licenses

Two sensors in the smart city YYY have to use API of open data portal in order to reuse data. Each dataset has a license hopefully belonging to Creative Commons, available online with the legal valid text and serialized in RDF using CC RDF metadata[1]. The Smart Contract aims to execute the Data Agreement between the two sensors. Before to start an API check if the dataset has the compatible licenses and this module passes to the smart contract the information about the permissions allowed.
The API module tests the ccREL assertion in RDF or in RDFa in the HTML text and pass to the Smart Contract the common vocabulary of creative commons. In the following example we have:

```
<?xml version="1.0" encoding="utf-8"?>
<rdf:RDF
   xmlns:cc="http://creativecommons.org/ns#"
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
     <rdf:Description
rdf:about="http://creativecommons.org/licenses/by/3.0/">
        <cc:requires rdf:resource="http://creativecommons.org/ns#Notice"/>
        <cc:requires
rdf:resource="http://creativecommons.org/ns#Attribution"/>
```

[1] https://creativecommons.org/ns http://www.w3.org/Submission/ccREL/ https://github.com/tdwg/rdf/blob/master/LicenseProperties.md

```
        <cc:permits
rdf:resource="http://creativecommons.org/ns#Distribution"/>

        <cc:permits
rdf:resource="http://creativecommons.org/ns#Reproduction"/>

        <cc:permits
rdf:resource="http://creativecommons.org/ns#DerivativeWorks"/>

        <cc:jurisdiction rdf:resource="http://dbpedia.org/resource/Spain"/>

   </rdf:Description>
</rdf:RDF>
```

But what is happening if the dataset applies a different license? How to harmonize the plenty of licenses used in open data? We need a common vocabulary and a mapping mechanism in order to provide to the smart contract a unique terminology to use.

4. State of the art

4.1 Smart Contract Ontology

Several studies already implemented the ontology inside of the smart contract but in order to standardize the steps inside of the transaction, the consensus algorithms, the repository. Ethereum and Ontology Smart Contract or smart contract template (DiMatteo 2019) already implemented some interesting results. Those ontologies are focused to describe the internal mechanism of the smart contract in order to support interoperability between different smart contracts developed in different platform or in different languages. Those ontologies are complementary respect the legal ontologies that are focused on the legal concept of the domain regulated by the smart contract.

EthOn ontology is Ethereum ontology for modelling the messaging in harmonized and standardized manner in order to have the same semantic in all the steps of the automatic transaction.

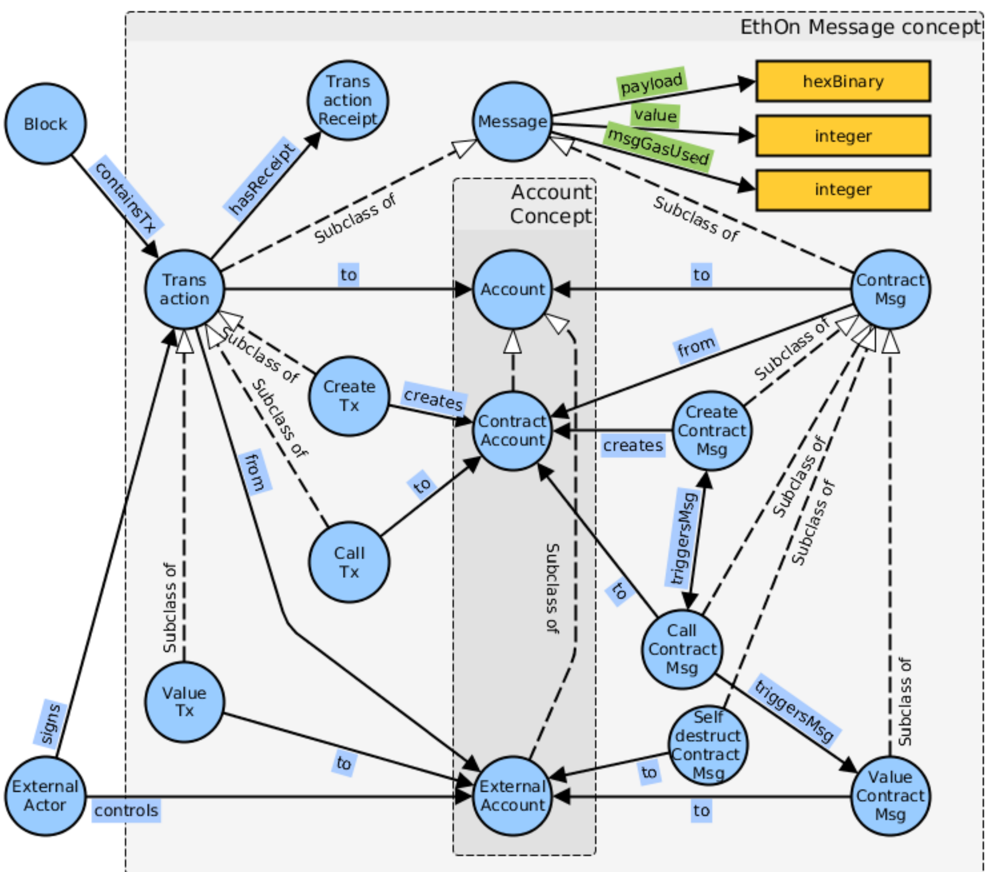


Figure 1 -EthOn Ontology.

Ontology Smart Contract is an ontology for modelling the transaction steps.

Smart Contract template, based on Ricardian contract method, provides a way to link standardised agreements to standardised code increasing the semantic richness. In (CLACK 2018) the authors stressed the point on the ambiguity and complexity of the legal norms, regulated by deontic operators. For this reason, the semantic of those concepts and their modelization in formal manner is extremely important in order to avoid misinterpretation.

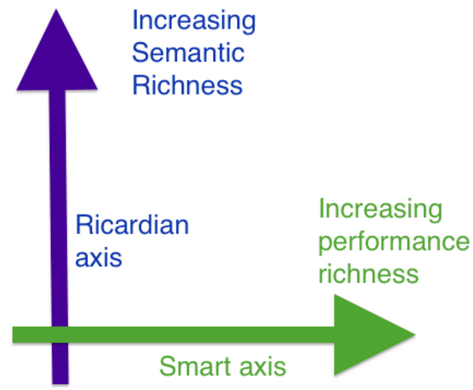


Figure 2 – Greeg model for increasing semantic

4.2 Other initiatives of standardization

Several groups in the standardization bodies are working on the standard for blockchain and smart contract (e.g., ISO, OASIS, CEN). In this perspective the definition of a common vocabulary, taxonomy and ontology is a part of their work.

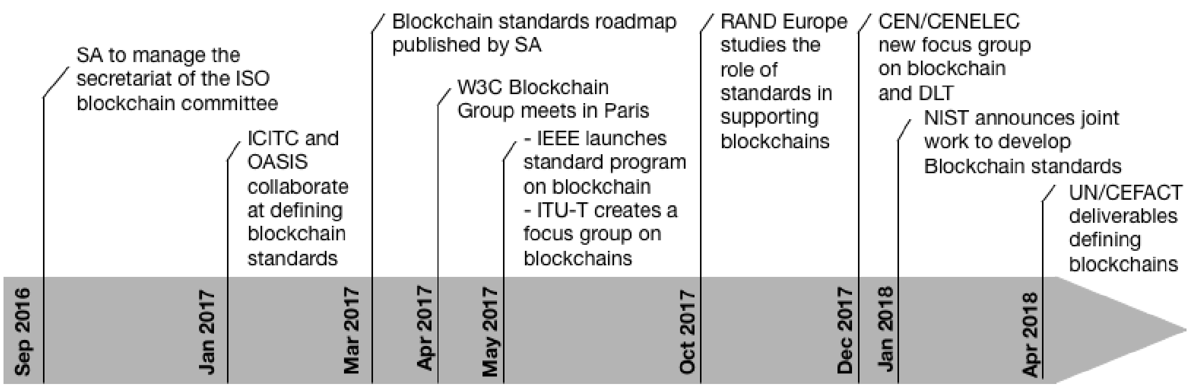


Figure 3 - Various organizations have started devoting effort to standardise blockchains and DLT (from GRAMOLI 2018)

We should add the recent (2019) open standard "Ethereum OASIS Open Project" that aims to create high-quality specifications that facilitate Ethereum's longevity, interoperability, and ease of integration. https://github.com/ethereum/oasis-open-project.

5. Pros and Cons Analysis

The main idea is the following:

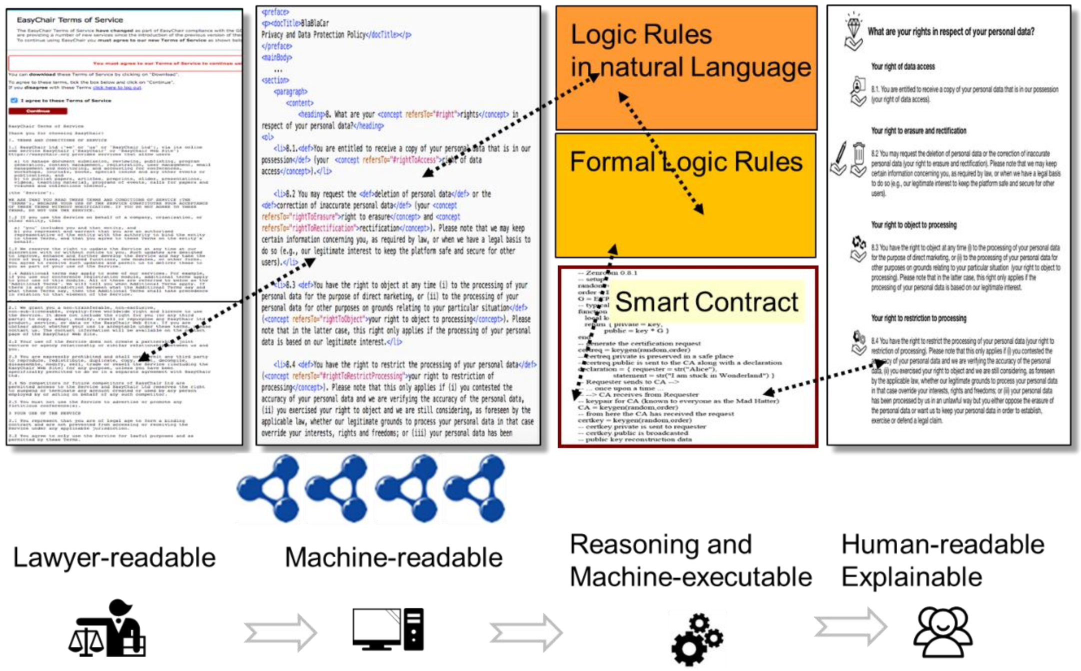


Figure 4 – Workflow of the legal knowledge modelling for a safe smart contract.

The legal text is transformed in machine-readable format enriched by legal ontology, then it is modelled in logic rules. After a checking compliance they are validated, and they could be converted in safe smart contract (Palmirani, Governatori 2018). Otherwise the risk is that the terminology used in the smart contract does not correspond to the legal concepts expressed in the legal text, and that the smart contract is not consistent with the legal rules based on deontic operators.

5.1 Advantages

We have different advantages to use legal ontology combined with the smart contracts scenario:

1. In case of modifications of the normative regulation, or of the application in different country (e.g., new municipality) or scenario (e.g., different geo-area) the smart contracts could be easily adapted using the connected local ontology and the knowledge graph, like a database, but more flexible and extensible.
2. Secondly, to use a common ontology permits to harmonize predicates and concepts especially in the IoT scenario, where different companies should cooperate each others through blockchain.
3. To have a robust ontology framework permits to communicate with the semantic web techniques (e.g., knowledge graphs).

5.2 Disadvantages

We have some disadvantages:

1. Legal ontology are really legal system dependent and language dependant. For this reason we will propose a meta-ontology.
2. To refine and populate an ontology in each different context is very hard. For this reason we provide a method for customizing the ontology to each concrete situation.
3. The ontology is an open word technology. Everything not described is considered unknown. The risk is to introduce a mechanism of inference that is less rigorous (approximate) respect the smart contract ecosystem. For this reason, we limited the ontology to objective assertions, simple Tbox and Abox, RDF serialization and knowledge graph technology.
4. To create Oracle/API from smart contract can produce a security breach or in any case the security should be reinforced.

6. Resulted expected

The results expected by the adoption of an ontology in the DECODE are the following:

1. to provide an **ontology methodology** in order to define **a DECODE ontology** to support the smart contracts application. The ontology should improve the flexibility and the customization of the smart contracts in situation where the parameters depends to external variables depending, often, to normative regulation.
2. to provide a little **proof-of-concept** with an example of integration between a fragment of ontology in zenroom with the contribution of Dyne.
3. to have a mechanism inside of the DECODE ontology for providing **explanation** of the data used in the smart contract process (***explicability***) in order to support transparency in the communication with the end-user according to the **GDPR principles**.

7. Methodology for building Legal Ontology

We have different level of ontologies according to the goal that the modeller of data requires. We have also different methodology to proceed: top-down coming from theoretical modelization of the reality; botton-up from the language, society, fact; mixed of the two.

We can use the ontology for different purposes, and it is fundamental to understand the macro-goal in order to chose the correct methodology:

1. Improve the searchability of the online sources using RDF assertions.

```
http://codev2.cc/download+remix/Lessig-Codev2.pdf isWrittenBy
https://it.wikipedia.org/wiki/Lawrence_Lessig;


http://codev2.cc/download+remix/Lessig-Codev2.pdf hasTitle "Code v2";
http://codev2.cc/download+remix/Lessig-Codev2.pdf hasFormat "PDF";
```

```
http://codev2.cc/download+remix/Lessig-Codev2.pdf hasLicense
http://creativecommons.org/licenses/by-sa/2.5/.
```

We can ask: «give me the license of the source titled "Code v2".» but we don't know nothing about the nature of the resource Code V2. It could be a journal paper or a book or simply a web page.

2. Improve the searchability of the online sources using concept relationships. We use RDFs or OWL for deducting new information about the connections between concepts:

```
http://codev2.cc/download+remix/Lessig-Codev2.pdf is_a
http://purl.org/dc/terms/BibliographicResource[2];

http://codev2.cc/download+remix/Lessig-Codev2.pdf is_a
http://purl.org/dc/terms/FileFormat;

https://it.wikipedia.org/wiki/Lawrence_Lessig is_a
http://purl.org/dc/terms/Agent;

http://codev2.cc/download+remix/Lessig-Codev2.pdf dc:issued "2006-01-01".
```

We now can ask more abstract query using classes: «give me all the book in digital format from "Lessig" as agent, issued in 2006».

3. Infer new information using OWL relationships.

```
http://codev2.cc/download+remix/Lessig-Codev2.pdf is_a
http://purl.org/dc/terms/BibliographicResource[3];

http://codev2.cc/download+remix/Lessig-Codev2.pdf is_a
http://purl.org/dc/terms/FileFormat;

https://it.wikipedia.org/wiki/Lawrence_Lessig is_a
http://purl.org/dc/terms/Agent;

http://codev2.cc/download+remix/Lessig-Codev2.pdf dc:issued "2006-01-01";

https://it.wikipedia.org/wiki/Lawrence_Lessig playsRole "Professor"

https://it.wikipedia.org/wiki/Lawrence_Lessig teaches "HarvardUniversity"

https://it.wikipedia.org/wiki/Lawrence_Lessig hasCourse "ICT Law"
```

We now can ask complex query: «give me all the courses that Lessig had in the period <t1-t2>»

[2] https://www.dublincore.org/specifications/dublin-core/dcmi-terms/

[3] https://www.dublincore.org/specifications/dublin-core/dcmi-terms/

4. Support reasoning using sophisticated axioms included in the OWL or preparing the work for logic reasoning or other logic programming application, including smart contract.

7.1 Level of knowledge modelling for legal sources

7.1.1 Foundational ontology

This class of ontology is so called *upper ontology* and the goal is model the fundamental classes of the reality using a philosophical top-down approach. Some very famous Upper ontology are Basic Formal Ontology (BFO[4]) and (DOLCE[5]). The role of those ontologies is to check if the modelization of the low-level concept is well done in order to avoid mistakes. It is typical the mistake between date and event: a date is an attribute of an event; in case an event could continue in interval of time. Another typical mistake is to confuse agent and role plaid by the agent. Those ontologies help the quality check of the concrete ontology.

7.1.2 Core ontology

The core ontologies are specific ontology by domain (e.g., mathematic, chemical, legal domain) where some general concept are defined. In Legal domain we have several good core ontologies. One of the most popular is LKIF ontology, used also by The National Archives of UK. Another very specific core ontology regards Deontic aspects of the norms (rights, permission, penalties, etc.). ODRL (Steyskal 2014) provides predicates and classes for managing obligations, permission, prohibitions, but several parts of the deontic logic are missing (e.g., right and penalty classes). LegalRuleML (Athan 2013, 2015) aims to provide also an ontology about those deontic aspects, and PrOnto developed a module for enriching the relationships between deontic operators for better modelling norms.

7.1.3 Domain ontology

The domain ontologies are specific for one branch of a domain. In legal domain we find GoodRelations (Distinto 2015) for eCommerce, PrOnto ontology for the privacy (Palmirani 2018, 2019), IRPOnto for the IPR domain.

7.1.4 Lexicon Ontology

For managing linguistic portion of the text like part-of-speech we need lexicon ontologies capable to detect verbs, syntagma, and part of the grammar.
Controlled vocabularies, thesauri and lexical databases are some examples of linguistic ontologies. They express the terminology concerning a domain of interest by organizing terms according to few semantic relations (e.g. hierarchical and associative ones). EUROC[6] and IATE[7] are some examples of linguistic ontologies released by the European Union to semantically structure the terminology of documents issued by EU institutions and bodies. However, these resources do not clarify the distinction between legal concepts and their instances.

[4] https://basic-formal-ontology.org/

[5] http://WWW.LOA.ISTC.CNR.IT/OLD/PAPERS/DOLCE-EKAW.PDF

[6] https://publications.europa.eu/en/web/eu-vocabularies/th-dataset/-/resource/dataset/eurovoc

[7] https://iate.europa.eu/

By contrast, the legal domain requires the modelling of legal core concepts, capable to overcome the vagueness of legal jargon that makes the meaning of legal terms subject to interpretation (Breuker 2006, 2007; Casellas 2011; Casanovas 2016)**Errore. L'origine riferimento non è stata trovata.**. Thus, the modelling of legal core ontologies is a complex task involving knowledge grounded on legal theory, legal doctrine and legal sociology.

Several models have been proposed as natural language interfaces to fill the gap between the high-level ontological concepts and their low-level, context-dependent lexicalisations (McCrae 2011, Leone 2019).
In particular, interesting works about SKOS-XL[8] and OntoLex are included in this version of PrOnto for combining ontology and linguistic literal forms, in support to NLP and search engine.

Lemon framework[9] is one of the most mature model for modelling lexicon ontology.

```
     :right a ontolex:LexicalEntry ;
        ontolex:sense :right_n_sense_1, :right_a_sense_2 ;
        ontolex:canonicalForm :right_n_form ;
        ontolex:otherForm :rights_n_form .

     : right_n_form a ontolex:Form ;
        ontolex:writtenRep "right"@en, "diritto"@it ;
        lexinfo:number lexinfo:singular .

     : rights_n_form a ontolex:Form ;
        ontolex:writtenRep "rights"@en, "diritti"@it;
        lexinfo:number lexinfo:plural .

     :right _n_sense_1 a ontolex:LexicalSense ;
        ontolex:isLexicalizedSenseOf :air_n_sense_1_lc .
        lexicog:restrictedTo :air_n_formRes .

     :right_a_sense_2 a ontolex:LexicalSense ;
        ontolex:isLexicalizedSenseOf :air_a_sense_2_lc ;

     :right_n_formRes a lexicog:FormRestriction ;
        lexinfo:number lexinfo:plural .

     :right_n_sense_1_lc a ontolex:LexicalConcept ;
        skos:definition  "Rights are entitlements (not) to perform certain
actions, or (not) to be in certain states; or entitlements that others (not)
perform certain actions or (not) be in certain states."@en .

     :right_a_sense_2_lc a ontolex:LexicalConcept ;
        skos:definition "just, fair, correct "@en .
```

8 https://www.w3.org/TR/skos-reference/skos-xl.html

9 https://jogracia.github.io/ontolex-lexicog/

7.1.5 Thesauri

Legal ontology needs to meet the text especially if you have different linguistic variants coming from different context (e.g., legal tradition, language, business logic scenarios). We can use SKOS for creating a common terminology and relationships between terms. The use of SKOS can be enough for creating a linguistic controlled vocabulary tracking also all the possible variants.

```
PrOnto:Controller rdf:type owl:Class;
rdfs:subClassOf PrOnto:Role;
rdfs:subClassOf skos:Concept.

PrOnto:DataController rdf:type PrOnto:Controller;
skosxl:prefLabel PrOnto:controller_1;
skosxl:altLabel               PrOnto:altController_1,PrOnto:altController_2,
PrOnto:altController_3, PrOnto:altController_4;

PrOnto:controller_1 rdf:type skosxl:Label;

skosxl:literalForm "controller"@en;
dct: created "2018-05-28"^^xsd:date;
dct: modified "2019-09-15"^^xsd:date.

PrOnto:altcontroller_1 rdf:type skosxl:Label;
skosxl:literalForm "data controller"@en.

PrOnto:altcontroller_2 rdf:type skosxl:Label;
skosxl:literalForm "company data controller"@en.

PrOnto:altController_3 rdf:type skosxl:Label;

skosxl:literalForm "company that is responsible for your information"@en.

PrOnto:altController_4 rdf:type skosxl:Label;
skosxl:literalForm "person responsible for processing"@en.
```

7.2 MeLOn (Methodology for building Legal Ontology)

The methodology is based on MeLOn (Methodology for building Legal Ontology) method. It is a button-up method with a foundational checking with the foundational ontologies (e.g., Dolce, OBL):

1. Describe the goal of the ontology (storytelling)
2. Evaluation indicators and parameters/indicators to evaluate the ontology
3. State of the art survey and other existing domain vocabularies
4. List all the relevant terminology and produce a glossary
5. Use tables to model the knowledge-base of the legal domain (excel)
6. Transform the tables in UML model using the Graffo tool
7. Transform the UML into OWL/XML serialization
8. Test the output under the technical and legal point of view (SPARQL queries on individuals)
9. Refine and optimize OWL by ontologist experts
10. Evaluate the ontology using the OntoClean method and goto 2)
11. Publish the document with the LODE tool and github

12. Collect feedbacks from the community and validation.

MeLOn is explicitly designed for legal ontologies and the related difficulties encountered by the legal operators during the definition of a model of reality through ontological techniques, such as Protégé, or patterns design method or the foundational approach. The MeLOn methodology evaluation indicators. PrOnto's criteria, based on the existing state of the art, are [6]: (i) coherence, (ii) completeness, (iii) efficiency, (iv) effectiveness, (v) usability, (vi) agreement. The MeLOn methodology allows to successfully work within interdisciplinary group that include engineers, lawyers, linguists, logicians and ontologists, and to model the legal knowledge rapidly and accurately while integrating the contributions of different disciplines. See also (Bandeira 2016) for other evaluation parameters.

8. Legal Ontology for DECODE

Inside of Decode we want to see how the legal ontology can support the smart contract modelling. We suggest this architecture, in case with also the possibility to have API/Oracle for collecting more information like in the approach (Liu 2019).

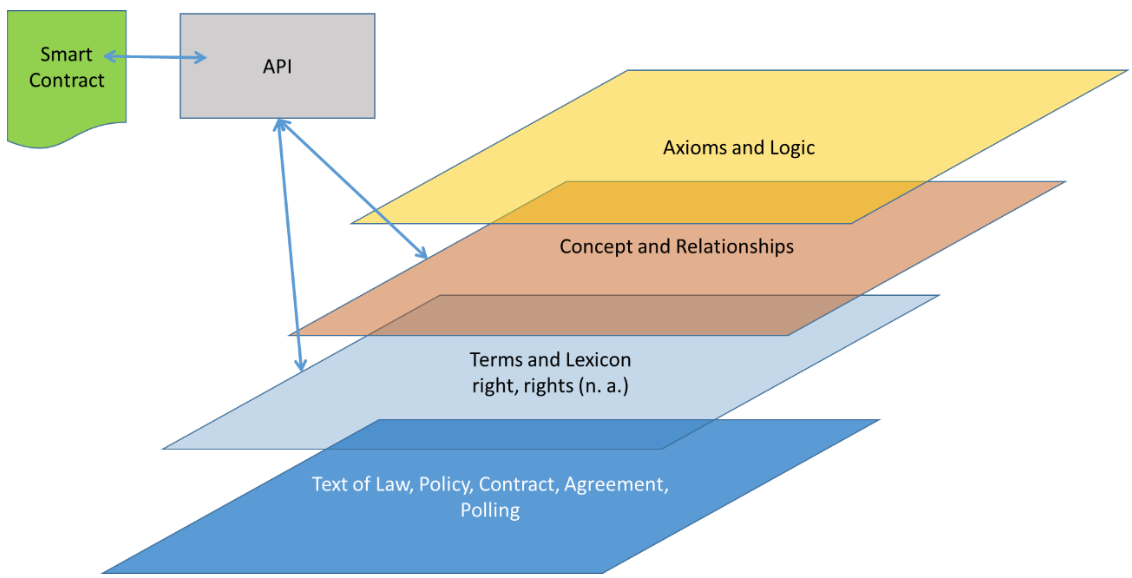


Figure 5 – Architecture for supporting smart contract in legal domain.

We need inputs from the text of legal norms, case-law, contracts, legal sources in general in order to extract the knowledge. From this level we extract knowledge using NLP and ML tools. Then we use those information mapped with already existing legal lexicon (e.g., Eurovoc) in order to favour the correct semantic assignment to the legal terms including multiple meaning (polysemy) and the hyponymy/hypernymy hierarchy (e.g., person is an hyperonym of child, right to be forgotten is a hypernym of the general class right). Then we can map the concept with a domain ontology (e.g., PrOnto) in order to discovery other relationship (e.g., controller and data subject relationships). This dialogue with the ontological level permits also to integrate the smart contract generally designed (e.g., template) with specific requirements according to the given situation (e.g., service of eGov, jurisdiction). Finally, we can use legal reasoning representation for checking the robustness of the rules before to transform them in smart contract (Baqa 2019).

9. Legal Ontology Building Blocks

The step number three of MeLOn methodology investigates the state of the art in order to reuse as much as possible the existing ontology patterns. The ontology design patterns is a methodology (Hitzler 2016).

ALLOT: this ontology implements the Akoma Ntoso Top Level Classes (TLCs) as a formal OWL 2 DL and allows to connect the data and document classes with the FRBR ontology (Barabucci et al., 2010).

FRBR: FRBR is an ontology that implements the FRBR model (IFLA Study Group on the Functional Requirements for Bibliographic Records, 1996). In particular we can re-use a subset of FRBR called FaBiO ontology is the optimization of FRBR ontology for modelling rights and metadata of physical and digital objects: https://sparontologies.github.io/fabio/current/fabio.html

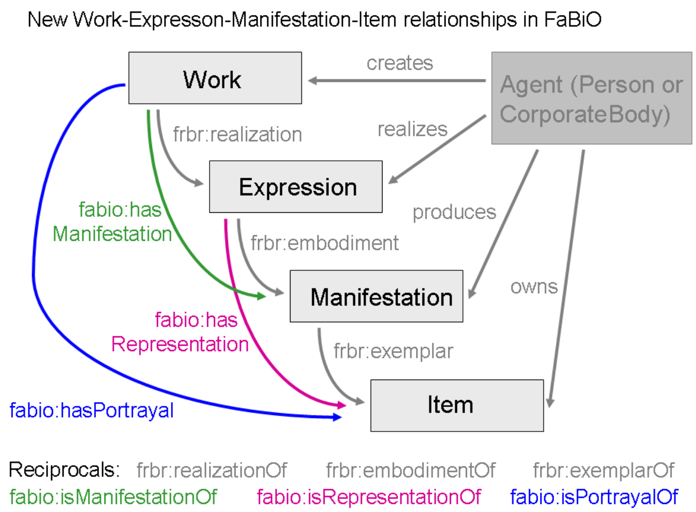


Figure 6 – FaBiO ontology.

LKIF Core: Action.owl is an ontology that represents actions in general, i.e., processes that are performed by an agent. We use in particular lkif:Agent to model lkif:Organization and lkif:Person (Breuker et al., 2007).

LKIF Core: Role.owl is an ontology to describe typologies of roles (epistemic roles, functions, person roles, organisation roles). We use in particular lkif:Role (Breuker et al., 2007).

The **Publishing Workflow Ontology** (PWO) is a simple ontology written in OWL 2 DL for the characterization of the main stages in the workflow associated with the publication of a document (e.g., being written, under review, XML capture, page design, publication on the Web). We reuse the workflow pattern to model the different types of processing of personal data (Gangemi et al., 2017).

Time-indexed Value in Context (TVC) is an ontology pattern that allows to describe scenarios in which someone (e.g., a person) has a value (e.g., a particular role) during a particular time and for a particular context. We use this portion of ontology to connect the event with value, context and time parameters (Peroni et al., 2017).

Time Interval (TI) is an ontology design pattern that enables the description of periods of time that are characterised by a starting date and an ending date. We use this ontology to manage the time interval (Peroni et al., 2017).

GeoNames is an ontology about the 11 million geonames toponyms, now with a unique URL and corresponding RDF web service. Other services describe the relation between toponyms like neighbours, nearby, children. The Ontology for GeoNames is available in OWL : http://www.geonames.org/ontology/ontology_v3.1.rdf.

LegalRuleML metadata model for the legal rules (Athan et al., 2013, 2015).
GoodRelations ontology of eCommerce http://www.heppnetz.de/projects/goodrelations/ (Distinto 2015).

RPaM Ontology https://github.com/everis-rpam/RPaM-Ontology/wiki/Ontology-Development-Report
https://joinup.ec.europa.eu/release/rpam-ontology/110
The RPaM (Representation of Powers and Mandates) Ontology is mainly based on the EU Directives, the General Data Protection Regulation (GDPR), the Regulation on electronic identification and trust services for electronic transactions in the internal market (eIDAS), and other EU legal framework related to the scope of the project. In RPaM are re-used all the ISA2 Core Vocabularies (e.g. Core Person Vocabulary, Core Public Service Vocabulary, other). RPaM is particular useful for defining powers and delegation chain between institutions (e.g., the son is delegated to manage the mother health record, the Authority is delegated by the Ministry to manage some action). Additionally there is a common vocabulary concerning the event of the life for citizens (person life events) and for companies (business life events).

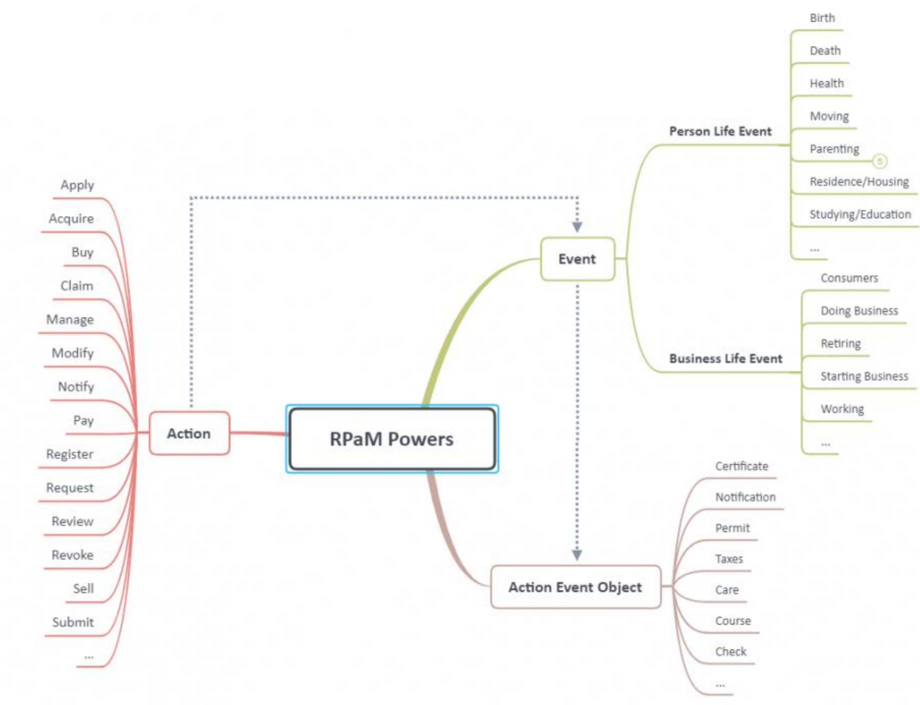


Figure 7 – RpaM ontology macro schema.

10. Legal Ontology about IPR and Commons

There are many legal ontologies for modelling Intellectual property rights: IPROnto, MPEG-21 Rights Data Dictionary and Rights Expression Languages, Creative Commons metamodel. Several tools (http://licentia.inria.fr/) provide also instruments for managing the checking compliance between different licenses.

The creative commons metamodel produces three levels of instruments: official legal text, icons for a short and effective communication, RDF triples with the information machine-readable. A similar model we can use for the smart contract.

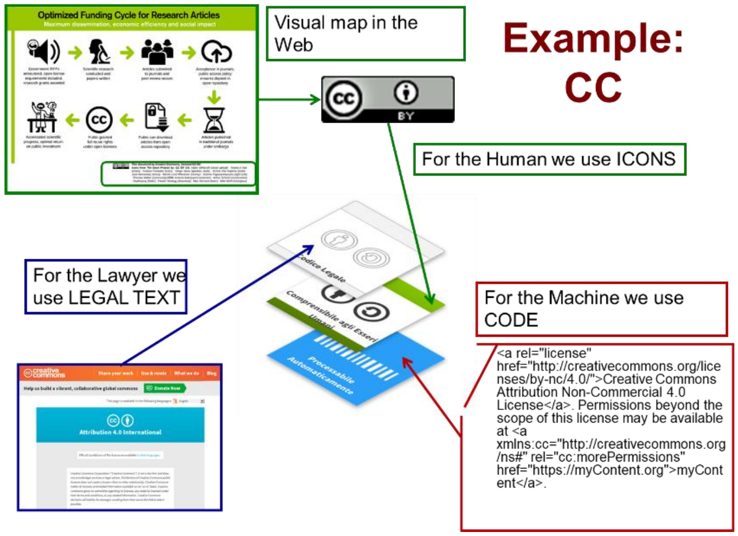


Figure 8 – Three level modelling of the legal knowledge: official legal license, icons, RDF.

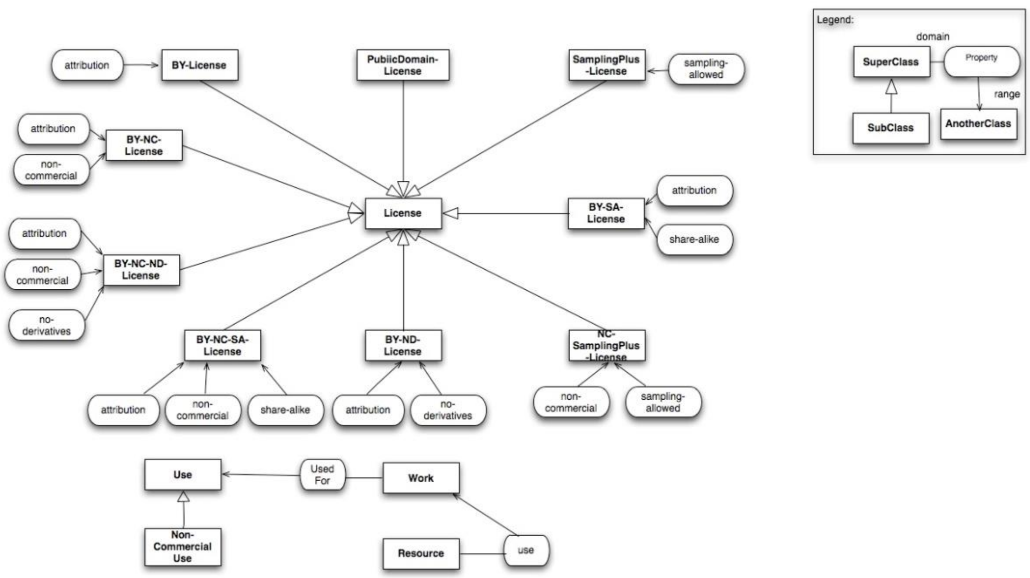


Figure 9 – Creative commons ontology.

We have analysed legal domain ontologies concerning IPR in order to see how this layer can help the DECODE goals. However, in DECODE there is a strong concept of *commonification* (BROUMAS 2019) governance of dataset as digital commons. In several deliverables (D1.9 Licensing of digital commons including personal data; D2.4 Data driven disruptive commons-based models; D1.8 Legal Framework for digital commons DECODE OS Legal Guidelines) there are the conceptualization of the digital commons. For this reason, we have modelled a very draft ontology of those concepts because never exist in the state of the art.

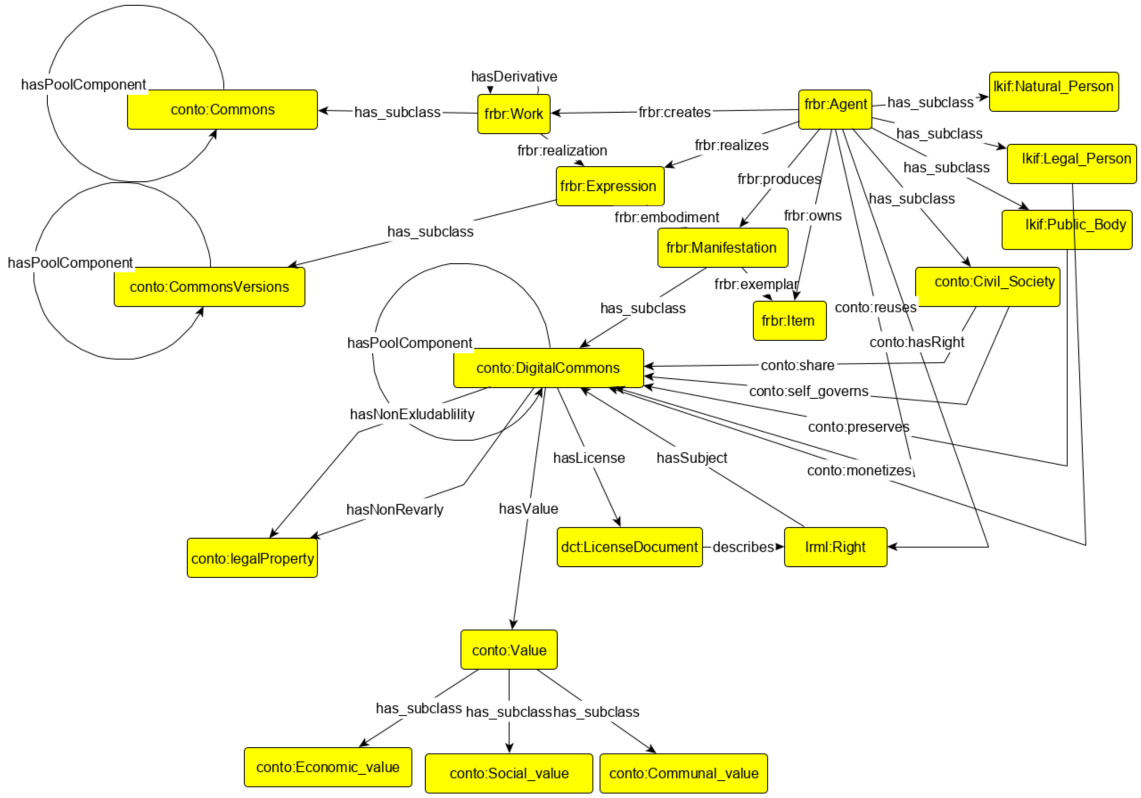


Figure 10 – COnto Commons Ontology.

The modelization of the commons ontology includes five main blocks:

- **FRBR** for modelling data and documents in their lifecycle (work, expression, manifestation, item);
- **Agent** that has some rights (lrlm:Right) and makes some action (e.g., create);
- **Commons** is a sub-class of FRBR in each step of the lifecycle, but it also could be a pool of other commons. Commons have two main legal properties: **nonrevality** and **nonexcludability**.
- Commons has **values**: economic, social, communal.
- Commons are **self-governed** by the Civil Society, but preserved by a Public body and reused by all.
- Commons have **license**, have **legal rights**.
- Legal person can **monetize** the commons.

11. Legal Ontology about Data Protection and Privacy

There are a lot of legal ontology concerning the data protection and privacy domain. UsablePrivacy and PrivOnto (Palmirani 2013, 2015) are ontologies oriented to provide linguistic tools in order to define glossary and taxonomy for the privacy domain, basically starting from the bottom-up annotation of the privacy policies (crowdsourcing annotation).

GDPRtEXT (Pandit 2018) lists concepts present in the GDPR text without really entering the modelling of the norms and the legal axioms (e.g., the actions performed by the processor, the obligations of the controller and the rights of the data subject). GDPRov aims to describe the provenance of the consent and data lifecycle in the light of the Linked Open Data principles such as Fairness and Trust. GConsent is an ontology for modelling the consent action, statement and actors. The SPECIAL Project[10] develops tools for checking compliance in privacy domain.
Finally, it is particular interest the common vocabulary "DPVCG GDPR" http://www.w3.org/ns/dpv-gdpr

12. Integration of Legal Ontology methodology in Zenroom

The Zenroom architecture presented in the D3.10 is the following:

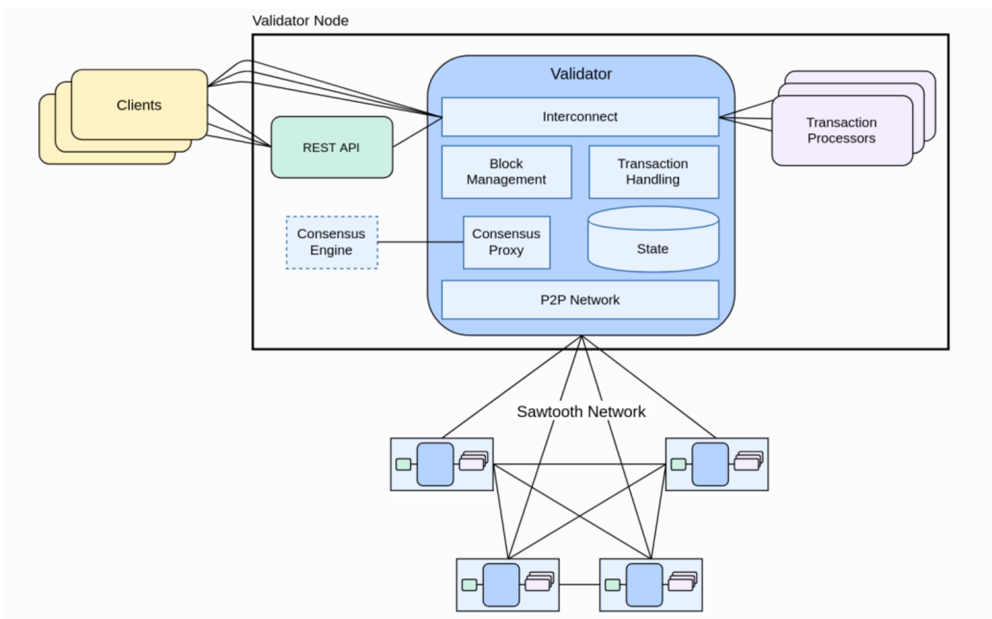


Figure 11 - Zenroom Architecture

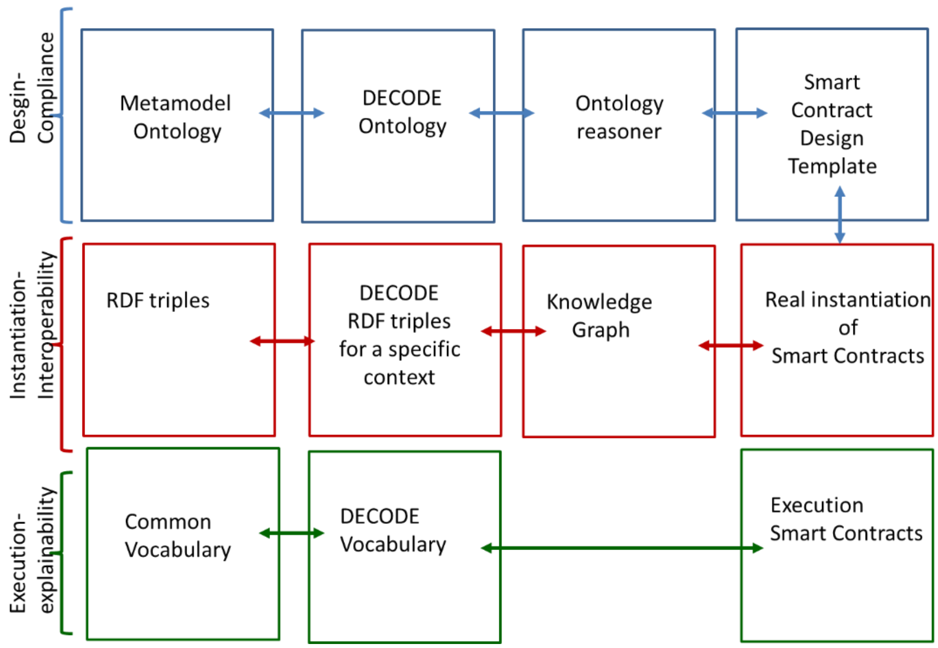
Actually we have three phases of possible integration between Zenroom and the legal ontologies:

1. Design of the smart contracts in order to make them compliant with the rules of the ontology (e.g., concept of citizenship, quorum, etc.). This is a meta-data model of the concept involved in the smart contracts creating *smart contracts templates*. We use the ontology Tbox for modelling IPR, PRIVACY, etc. legal concepts.
2. Instantiation of the real facts/data coming from the populated ontology into the smart contracts templates in order to materialize a concrete instance of the smart

[10] https://www.specialprivacy.eu/

contracts. We use facts and Abox coming from RDF assertions to populate the smart contracts.

3. Implementation of smart contracts pipeline: we use the ontology Tbox for implementing the *explanation* principle.



1. Figure 12 – General framework for DECODE ontology.

This picture shows the use of the ontology and of the common vocabulary in the DECODE perspective.

The metamodel legal ontology (TBox axioms) helps the design step of the smart contract general for a specific situation, but not customized. The ontology permits to easily customize the generic smart contract with peculiar requirements. The RDF triples provide real data for the instantiation of the smart contract during the real execution. The common vocabulary and the legal ontology relationships can explain the output of the execution for implementing the right to explain and the right of know (art. 22, recital 71 of GPDR, High-Level Expert Group on AI guidelines)

However the legal ontology is not enough for modelling legal norms. Exceptions are frequently used in public administration scenario (eGov) and in the regulation. Moreover, the smart contracts are written in different languages, for different platform, following different methodology and method of execution. For this reason a legal ontology can help to harmonize the usage of the same legal concepts and to improve the legal rule modelling.

13. Example

Suppose to have a term of use for “Massager kids by XXX” and you want to model it in order to produce a smart contract that in case of deletion of the account it is automatically deleted all the cloud computing

"Messenger **Kid**s by xxx is a children's messaging and video calling app that helps your children communicate with family and friends in a fun, controlled environment. This privacy policy explains what information we collect from your **child** when they use Messenger Kids and how we use and share that information."
The rule to automatize is:

If you delete your child's account, ***(then)*** we must delete their Messenger Kids information.

```
PrOnto:Child rdf:type owl:Class;
rdfs:subClassOf PrOnto:Role;
rdfs:subClassOf skos:Concept.
PrOnto:ChildGDPR rdf:type PrOnto:Child;
skosxl:prefLabel "child";
skosxl:altLabel  "children", "kid", "kids"
```

Using PrOnto:Child class ontology we infer new knowledge:

```
PrOnto:Child is_a PrOnto:DataSubject
PrOnto:DataSubject hasRigh PrOnto:RightToErasure
PrOnto:RightToErasure generates PrOnto:ObligationToErase
```

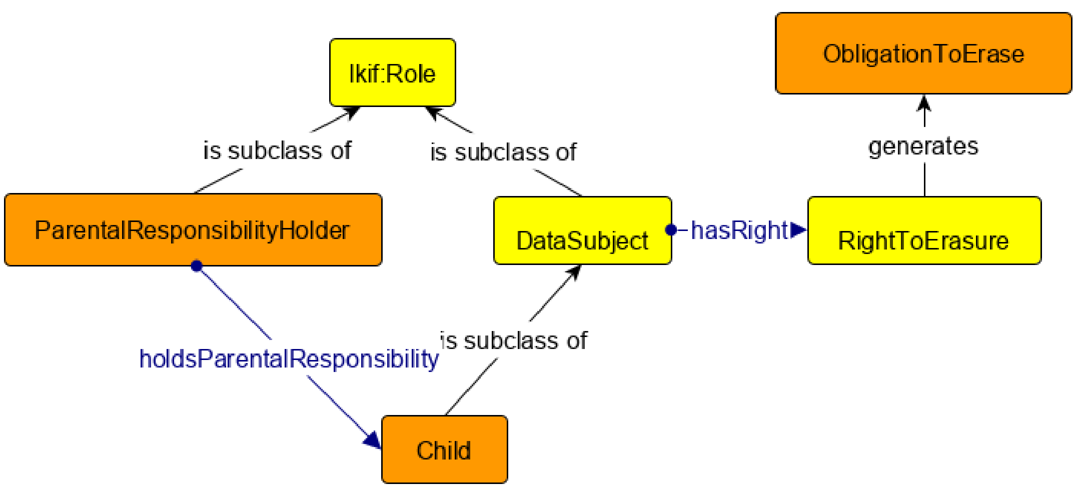


Figure 13 – Fragment of the PrOnto ontology.

At the smart contract level we check the context conditions.

```
IF
      AND
      X is_a PrOnto:Person
      X has Y yearsOld
      X belongs to J
THEN
      X playRoleOf PrOnto:Child
```

And also the rules to delete.

```
IF
      X PrOnto:deletes D
      C is_a PrOnto:Controller
THEN
OBLIGATION
      C PrOnto:deletes D
```

The PrOnto ontology provides the use of standard classes (`PrOnto:Controller`, `PrOnto:Child`) and predicates (`PrOnto:deletes`). We can also recall the relationships between controller and child in case we have further constraints modelled inside of the ontology (e.g., parentships relationships) and using those information we can reinforce the smart contract.

Bibliography

[1] Ashley, K.D., 2017. Artificial intelligence and legal analytics: new tools for law practice in the digital age. Cambridge Univ Press, Cambridge New York Melbourne Delhi Singapore.

[2] Athan T, Governatori G, Palmirani M, Paschke A, Wyner A (2015) LegalRuleML: design principles and foundations. In: Reasoning web international summer school. Springer, Berlin, pp 151–188

[3] Athan, T., Harold Boley; Guido Governatori; Monica Palmirani; Adrian Paschke; Adam Wyner, (2013) LegalRuleML: from Metamodel to Use Cases - A Tutorial, in: RuleML2013 - Theory, Practice, and Applications of Rules on the Web, Springer Berlin Heidelberg, 2013, pp. 1 - 6 (atti di: RuleML2013, Seattle, 11-13 July, 2013).

[4] Bandeira, J., Bittencourt, I.I., Espinheira, P., Isotani, S., (2016). FOCA: A Methodology for Ontology Evaluation. Eprint ArXiv.

[5] Baqa H., N. B. Truong, N. Crespi, G. M. Lee and F. le Gall, Semantic Smart Contracts for Blockchain-based Services in the Internet of Things, (2019) IEEE 18th International Symposium on Network Computing and Applications (NCA), Cambridge, MA, USA, 2019, pp. 1-5.

[6] Barabucci, G., Cervone, L., Di Iorio, A., Palmirani, M., Peroni, S., Vitali, F., 2010. Managing semantics in XML vocabularies: an experience in the legal and legislative domain. In Proceedings of Balisage: The markup conference (Vol. 5).

[7] Bosque-Gil, J., Gracia, J., Montiel-Ponsoda E., (2017). Towards a module for lexicography in OntoLex. In Proc. of the LDK workshops: OntoLex, TIAD and Challenges for Wordnets at 1st Language Data and Knowledge conference (LDK 2017), Galway, Ireland, vol. 1899. CEUR-WS, 2017, pp. 74-84.

[8] Breuker J, Hoekstra R (2004) Epistemology and ontology in core ontologies: FOLaw and LRI-Core, two core ontologies for law. In: Proceedings of the EKAW04 workshop on core ontologies in ontology engineering. Northamptonshire, UK, pp 15–27

[9] Breuker, J., Hoekstra, R., Boer, A., van den Berg, K., Sartor, G., Rubino, R., Wyner, A., Bench-Capon, T., Palmirani, M., (2007). OWL Ontology of Basic Legal Concepts (LKIF-Core), Deliverable No. 1.4.

IST-2004-027655 ESTRELLA: European project for Standardised Transparent Representations in order to Extend Legal Accessibility.

[10]BROUMAS A, (2019), The Ontology of the Intellectual Commons International Journal of Communication 11(2017), 1507–1527

[11]Casanovas, Pompeu; Palmirani, Monica; Peroni, Silvio; Van Engers, Tom; Vitali, Fabio, (2016) *Semantic Web for the Legal Domain: The'next step*, «SEMANTIC WEB», 2016, 7, pp. 213 – 227.

[12]Casellas, N., (2011) Legal ontology engineering: methodologies, modelling trends, and the ontology of professional judicial knowledge, vol 3. Springer, Berlin.

[13]Cer, D., Yang, Y., Kong, S. Y., Hua, N., Limtiaco, N., John, R. S., Sung, Y. H., (2018). Universal sentence encoder. arXiv preprint arXiv:1803.11175.

[14]Clack, C. D., Bakshi, V. A., & Braine, L. (2016). Smart contract templates: foundations, design landscape and research directions. arXiv preprint arXiv:1608.00771.

[15]Clack, Christopher. (2018). Smart Contract Templates: legal semantics and code validation, C. D. Clack. Journal of Digital Banking 2(4) 338-352. 2018.

[16]Contract Law and Smart Contracts. (2019). In L. DiMatteo, M. Cannarsa, & C. Poncibò (Eds.), The Cambridge Handbook of Smart Contracts, Blockchain Technology and Digital Platforms (Cambridge Law Handbooks, pp. 59-140). Cambridge: Cambridge University Press.

[17]de Oliveira Rodrigues CM, de Freitas FLG, Barreiros EFS, de Azevedo RR, de Almeida Filho AT (2019) Legal ontologies over time: a systematic mapping study. Expert Syst Appl 130:12–30.

[18]Declerck, T., Egorova, K., & Schnur, E., (2018). An Integrated Formal Representation for Terminological and Lexical Data included in Classification Schemes. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018).

[19]Del Corro, L., Gemulla, R., (2013). ClausIE: clause-based open information extraction. In: Proceedings of the 22nd international conference on World Wide Web. ACM, 2013, pp. 355-366.

[20]Distinto I, d'Aquin M, Motta E (2016) LOTED2: an ontology of european public procurement notices. Semant Web 7(3):267–293.

[21]ELI Task Force (2018) ELI implementation methodology: good practices and guidelines. Publications Office.

[22]Fernández-Barrera, M., & Sartor, G., (2011). The legal theory perspective: doctrinal conceptual systems vs. computational ontologies. In Approaches to Legal Ontologies. Springer, Dordrecht, pp. 15-47.

[23]Gandon F, Governatori G, Villata S (2017) Normative requirements as linked data. In: The 30th international conference on legal knowledge and information systems (JURIX 2017).

[24]Gangemi A, Presutti V (2009) Ontology design patterns. In: Handbook on ontologies. Springer, Berlin, pp 221–243.

[25]Gangemi A, Sagri M-T, Tiscornia D (2005) A constructive framework for legal ontologies. In: Law and the semantic web. Springer, Berlin, pp 97–124.

[26]Gangemi, A., Guarino, N., Masolo, C., Oltramari, A., Schneider, L., (2002). Sweetening Ontologies with DOLCE. In International Conference on Knowledge Engineering and Knowledge Management Springer, Berlin, Heidelberg, pp. 166-181.

[27]Gangemi, A., Peroni, S., Shotton, D., Vitali, F., (2017). The Publishing Workflow Ontology (PWO). Semantic Web 8, pp. 703–718. https://doi.org/10.3233/SW-160230

[28]Gangemi, A., Presutti, V., Reforgiato Recupero, D., Nuzzolese, A. G., Draicchio, F., & Mongiovì, M., (2017). Semantic web machine reading with FRED. Semantic Web, 8(6), pp .873-893.

[29]Gramoli V., M. Staples, (2018), Blockchain Standard: Can We Reach Consensus?, in IEEE Communications Standards Magazine, vol. 2, no. 3, pp. 16-21, SEPTEMBER 2018.

[30]Grigg, I (2015), On the intersection of ricardian and smart contracts. http://iang.org/papers/intersection_ricardian_smart.html.

[31]Guarino, N., Welty C.A., (2004). An Overview of OntoClean. In Handbook on ontologies. Springer, Berlin, Heidelberg, pp. 151-171.

[32]Hepp M. (2008) GoodRelations: An Ontology for Describing Products and Services Offers on the Web. In: Gangemi A., Euzenat J. (eds) Knowledge Engineering: Practice and Patterns. EKAW 2008. Lecture Notes in Computer Science, vol 5268. Springer, Berlin, Heidelberg.

[33]Hitzler, P., Gangemi, A., Janowicz, K., Krisnadhi, A. (Eds.), (2016). Ontology engineering with ontology design patterns: foundations and applications, Studies on the semantic web. IOS Press, Amsterdam Berlin.

[34]Hoekstra R, Breuker J, Di Bello M, Boer A et al (2007) The LKIF core ontology of basic legal concepts. LOAIT 321:43–63

[35]http://licentia.inria.fr/

[36]http://ontologydesignpatterns.org/wiki/Main_Page

[37]http://openscience.adaptcentre.ie/ontologies/GConsent/docs/ontology

[38]http://purl.org/NET/rdflicense

[39]http://www.loa.istc.cnr.it/old/Papers/DOLCE-EKAW.pdf

[40]http://www.w3.org/2016/05/ontolex

[41]https://www.w3.org/ns/dpv#data-controller

[42]IFLA Study Group on the Functional Requirements for Bibliographic Records, 1996. Functional Requirements for Bibliographic Records, IFLA Series on Bibliographic Control. De Gruyter Saur.

[43]Legal and regulatory framework of blockchains and smart contracts a thematic report prepared by The European Union Blockchain Observatory & Forum.

[44]Leone, V., Di Caro, L. & Villata, S. Taking stock of legal ontologies: a feature-based comparative analysis. Artif Intell Law (2019). https://doi.org/10.1007/s10506-019-09252-1

[45]Liu S., F. Mohsin, L. Xia and O. Seneviratne, Strengthening Smart Contracts to Handle Unexpected Situations, (2019), IEEE International Conference on Decentralized Applications and Infrastructures (DAPPCON), Newark, CA, USA, 2019, pp. 182-187.

[46]McCrae, J., Spohr, D., Cimiano, P., (2011). Linking Lexical Resources and Ontologies on the Semantic Web with Lemon. In: Antoniou G. et al. (eds) The Semantic Web: Research and Applications. ESWC 2011. Lecture Notes in Computer Science, vol 6643. Springer, Berlin, Heidelberg.

[47]Muñoz-Soro JF, Esteban G, Corcho O, Serón F (2016) PPROC, an ontology for transparency in public procurement. Semant Web 7(3):295–309.

[48]Oltramari, A., Piraviperumal, D., Schaub, F., Wilson, S., Cherivirala, S., Norton, T.B., Russell, N.C., Story, P., Reidenberg, J., Sadeh, N., (2016). Privonto: A semantic framework for the analysis of privacy policies. Semantic Web (1-19).

[49]Pagallo Ugo, Monica Palmirani, Pompeu Casanovas, Giovanni Sartor, Serena Villata, Introduction: Legal and Ethical Dimensions of AI, NorMAS, and the Web of Data, in: AI Approaches to the Complexity of Legal Systems, Berlin, Springer, (2018), pp. 14 – 33.

[50]Palmirani M, Governatori G, Rotolo A, Tabet S, Boley H, Paschke A (2011) Legalruleml: Xml-based rules and norms. In: Rule-based modeling and computing on the semantic web. Springer, Berlin, pp 298–312

[51]Palmirani Monica; Michele Martoni; Arianna Rossi; Cesare Bartolini; Livio Robaldo, (2018) PrOnto: Privacy Ontology for Legal Reasoning, in: Electronic Government and the Information Systems Perspective. 7th International Conference, EGOVIS 2018, Regensburg, Germany, September 3–5, 2018, Proceedings, Cham, Springer Nature Switzerland AG, «LECTURE NOTES IN COMPUTER SCIENCE», 2018, pp. 139 - 152 (atti di: EGOVIS - Electronic Government and the Information Systems Perspective, Regensburg, Germany, 2018 September 3-5).

[52]Palmirani, M., Martoni, M., Rossi, A., Bartolini, C., Robaldo, L., (2018). PrOnto: Privacy Ontology for Legal Reasoning. EGOVIS2018, 7th International Conference, EGOVIS 2018, Regensburg, Germany, September 3-5, 2018, Proceedings. LNCS 11032, Springer, pp. 139-152.

[53]Palmirani, M., Martoni, M., Rossi, A., Bartolini, C., Robaldo, L., 2018. Legal Ontology for Modelling GDPR Concepts and Norms. JURIX 2018. pp. 91-100.

[54]Palmirani, M., Martoni, M., Rossi, A., Bartolini, C., Robaldo, L., (2018). PrOnto: Privacy Ontology for Legal Compliance. In: Proceedings of the 18th European Conference on Digital Government ECDG 2018, Reading UK, Academic Conferences and Publishing International Limited, 2018, pp. 142 – 151.

[55]Palmirani, M.,Governatori, G., 2018. Modelling Legal Knowledge for GDPR Compliance Checking. JURIX 2018, pp. 101-110.

[56]Pandit HJ, Fatema K, O'Sullivan D, Lewis D (2018) GDPRtEXT-GDPR as a linked data resource. In: European semantic web conference. Springer, Berlin, pp 481–495

[57]Pandit, H.J., Fatema, K., O'Sullivan, D., Lewis, D., 2018. GDPRtEXT - GDPR as a Linked Data Resource. In Gangemi A. et al. (eds) The Semantic Web. ESWC 2018. Lecture Notes in Computer Science, vol 10843. Springer, Cham.

[58]Pandit, H.J., Lewis, D., 2017. Modelling Provenance for GDPR Compliance using Linked Open Data Vocabularies. In Proceedings of the 5th Workshop on Society, Privacy and the Semantic Web - Policy and Technology (PrivOn2017) co-located with the 16th International Semantic Web Conference (ISWC 2017), http://ceur-ws.org/Vol-1951/PrivOn2017_paper_6.pdf.

[59]Peroni, S., Palmirani, M., Vitali, F., 2017. UNDO: The United Nations System Document Ontology, in: d'Amato, C., Fernandez, M., Tamma, V., Lecue, F., Cudré-Mauroux, P., Sequeda, J., Lange, C., Heflin, J. (Eds.), The Semantic Web – ISWC 2017. Springer International Publishing, Cham, pp. 175–183. https://doi.org/10.1007/978-3-319-68204-4_18.

[60]Rossi, A., Palmirani, M., 2019. DaPIS: an Ontology-Based Data Protection Icon Set. In G. Peruginelli and S. Faro (Eds.): Knowledge of the Law in the Big Data Age. Frontiers in Artificial Intelligence and Applications. Vol. 317. IOS Press.

[61]Roussey, Catherine, Pinet, F., Kang, M. A., & Corcho, O., (2011). An introduction to ontologies and ontology engineering. In Ontologies in Urban development projects. Springer, London, pp. 9-38.

[62]Steyskal S, Polleres A (2014) Defining expressive access policies for linked data using the ODRL ontology 2.0. In: Proceedings of the 10th international conference on semantic systems, ACM, pp 20–23.
[63]Valente A, Breuker J et al (1994) A functional ontology of law. Towards a global expert system in law, 112–136.
[64]van Kralingen R (1997) A conceptual frame-based ontology for the law. In: Proceedings of the first international workshop on legal ontologies, pp 6–17.
[65]van Opijnen, M.and Santos, C., 2017. On the concept of relevance in legal information retrieval, Artificial Intelligence and Law 25: 65. https://doi.org/10.1007/s10506-017-9195-8