Journal of Theoretical and Applied Information Technology
31st May 2025. Vol.103. No.10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

TO-MULTILONTOLOGY & MPCO: A METHODOLOGY FOR DEVELOPING MULTILINGUAL ONTOLOGIES & A LEGAL ONTOLOGY OF THE PENAL CODE

ISMAHANE KOURTIN[1,2]

[1]Bourgogne-Franche-Comté University, CRIT Laboratory, Besançon, France

[2]Ibn Tofail University, Faculty of Science, EDPAGS Laboratory, Kenitra, Morocco

E-mail: [1]kourtin_ismahane.math@yahoo.fr

ABSTRACT

Ontologies are among the techniques introduced by artificial intelligence in the early 1990s to enable better organization and semantic representation of information. They have the potential to play a crucial role in the design of question-answering systems and content comprehension by organizing and structuring the data they present. Multilingual ontologies are both language-independent and capable of supporting multiple languages, offering significant potential for querying and understanding knowledge in multicultural and multilingual environments.

Although several ontology development methodologies exist, they provide the necessary elements for ontology construction without clearly demonstrating how to implement them or specifying the models to guide the development process, particularly for multilingual ontologies. Indeed, existing methodologies summarize the development of ontologies as a mere enumeration of important terms, followed by the definition of classes and their hierarchy, the definition of properties and their facets, and finally the creation of instances—without showing users the approach or method that could guide them in choosing terms, defining classes, the hierarchy, and properties, or in demonstrating how to build multilingual ontologies. In addition, there is a lack of models that allow for representing ontology data in a way that guides its development and documentation.

This article proposes a customized methodology, TO-MULTILONTOLOGY, which covers aspects from the specification phase to the validation and evaluation phase, offering a detailed implementation process with clearly defined steps to guide and simplify the task of building multilingual ontologies. The proposed methodology also addresses one of the main obstacles to effective knowledge sharing: the inadequate documentation of existing ontologies. It provides powerful tools and models that not only document the ontology but also guide its development. This methodology will be explained and applied in the development of a multilingual legal ontology, MPCO (Multilingual Penal Code Ontology), in French and Arabic, for the Moroccan government's Penal Code. The constructed ontology can play a significant role in information retrieval and in learning about the penal code. It can also serve as a reference for the development of similar penal law ontologies.

Keywords: *Legal Ontologies, Ontology Development, Ontology Design, Multilingual Ontologies, Knowledge Representation And Modeling, Ontology Construction And Development Methodologies.*

1. INTRODUCTION

The ever-growing mass of information has created a crucial need to organize and structure the contents of available documents, transforming them into an intelligent guide capable of providing comprehensive and immediate answers to natural language queries. Ontologies are a promising solution that continues to prove its effectiveness. They enable the structuring of data and the creation of meaningful links by leveraging semantic web technologies and standards. These ontological datasets can be queried using SPARQL.

Ontologies were introduced by artificial intelligence in the early 1990s to enable better organization and semantic representation of information. They have the potential to play a critical role in the design of question-answering

systems and content comprehension by structuring and organizing data. Multilingual ontologies offer even greater potential for querying and understanding knowledge in multicultural and multilingual environments, thanks to their language independence and ability to support multiple languages and linguistic variations.

Several methodologies have been proposed to guide the design and development of ontologies, but their implementation remains unclear and poorly suited for the development of multilingual ontologies. More specifically, these methodologies outline the steps necessary for building ontologies without showing how to implement them or specifying the models to guide their execution.

Despite differences among methodologies in the order of steps and in the language used to define them, they all adopt conceptually similar stages and requirements, which can be defined as: specification, conceptualization, implementation, evaluation, and documentation. Conceptualization is the main operation, consisting of identifying terms, grouping them into semantic classes, and structuring them into a terminological network. However, all these methodologies reduce ontology development to an enumeration of important terms, followed by the definition of classes and their hierarchy, the specification of properties and their facets, and finally the creation of instances—without providing users with a clear approach or guidance to help them choose the terms, define the classes, hierarchy, and properties, or show them how to build multilingual ontologies. Added to this is the lack of models to represent ontology data in a way that supports its development and documentation.

This article proposes a customized methodology, TO-MULTILONTOLOGY, which covers aspects from the specification phase to the validation and evaluation phase, with a detailed implementation process featuring clearly defined steps that guide and simplify the task of building multilingual ontologies. The proposed methodology also addresses one of the main obstacles to effective knowledge sharing: the inadequate documentation of existing ontologies. It provides powerful tools and models that document the ontology and guide its development.

This methodology will be explained and applied in the development of a multilingual legal ontology, MPCO (Multilingual Penal Code Ontology), in French and Arabic, for the Moroccan government's Penal Code. The constructed ontology can play a significant role in information retrieval and in learning about the penal code. It can also serve as a reference for the development of similar penal law ontologies.

The proposed methodology for the development of multilingual ontologies consists of seven steps:

1) Establishing the ontology charter — Specification;
2) Building the ontology skeleton;
3) Defining the basic properties between the core concepts of the ontology skeleton;
4) Conceptualizing and refining the ontology;
5) Identifying and creating individuals;
6) Verifying the ontology's consistency and simulating deductive reasoning;
7) Validating and evaluating the ontology.

The tool used for the construction and development of the ontology is Protégé 5.6.4, a free and open-source tool for editing and managing ontologies. Michael DeBellis has created a detailed guide on using Protégé version 5.5 for ontology development [1].

The rest of this article is organized as follows: Section 2 provides an overview of ontologies, covering a general introduction to ontologies, legal ontologies, methodologies, modeling languages, and existing tools for their development. Section 3 presents the TO-MULTILONTOLOGY methodology proposed for ontology construction, as well as the development of the multilingual legal ontology of the Moroccan Penal Code (MPCO). Finally, Section 4 concludes this work.

2. CONTEXT OF ONTOLOGIES

2.1 Introduction

The concept of ontology is a term borrowed from philosophy and repurposed as an IT object. For centuries, philosophers have attempted to classify things and analyze their properties to better understand the world around them. The Greek philosopher Plato (428-348 BCE) was already establishing categories based on fundamental questions about reality, existence, and the true nature of things. This branch of philosophy is now known as Ontology and is defined as "the study of being qua being," according to Aristotle's (384-322 BCE) definition, who was Plato's student [2].

Human reasoning is based on what can be called an ontology of the world, meaning a certain view of the world and the categories that organize it. The knowledge representation community

Journal of Theoretical and Applied Information Technology
31st May 2025. Vol.103. No.10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

adopted the term ontology in the 1990s to refer to the object resulting from a knowledge modeling process. The most widely accepted definition of an ontology is the one introduced by Gruber [3, 4] and extended by Borst [5], who defines an ontology as "a formal explicit specification of a shared conceptualization," where conceptualization refers to the objects, concepts, and other entities believed to exist within a particular domain of interest (the universe of discourse) and the relationships that exist between these entities. Studer et al. [6] elaborate on this definition: a "conceptualization" refers to an abstract model of a phenomenon in the world, having identified the relevant concepts of that phenomenon; "explicit" means that the types of concepts used and the constraints on their usage are explicitly defined; "formal" refers to the fact that the ontology must be machine-readable; and "shared" reflects the idea that an ontology captures consensual knowledge, meaning it is not private to an individual but accepted by a group. Such an ontology consists of a set of concepts that are both hierarchically organized and structured by relationships linking these concepts.

The article published in 1996 by Uschold and Gruninger [7] remains a foundational text on ontologies and the methodology for their construction, where ontology is defined as "a term used to refer to the shared understanding of a domain of interest that can be used as a unifying framework to solve problems of communication between people and interoperability between systems."

McGuinness et al. [8] also contributed to the clarification and development of the ontology, which they define as a formal and explicit description of the concepts in a domain of discourse (classes, sometimes referred to as concepts), the properties of each concept describing various characteristics and attributes of the concept (attributes for slots, sometimes called roles or properties), and restrictions on the slots (facets, sometimes called role restrictions). Ontologies, along with the set of individual instances of the classes, form a knowledge base.

An ontology as an IT artifact is therefore composed of concepts, the relationships between them, their definitions, their properties, constraints on the properties, and individuals. Figure 1 presents the constituent parts of an ontology.

The set of concepts, their definitions, and the relationships between them, represented hierarchically, is what is called a taxonomy. A taxonomy is a method of classifying or categorizing a set of things using a hierarchical structure, that is, a tree structure, with the most general category as the root of the tree. Each node, including the root node, is an information entity representing a real-world object that is being modeled. Each link between two nodes in a taxonomy represents a subcategorization or supercategorization relationship.

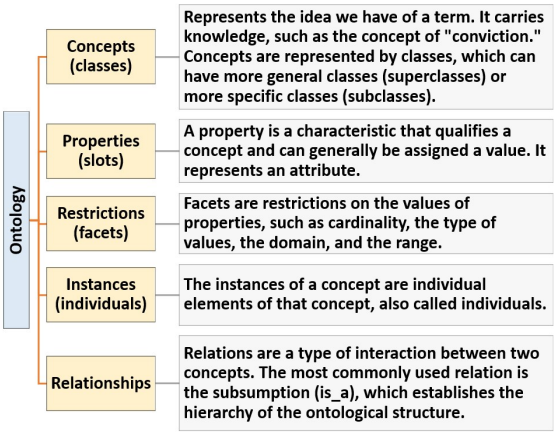


Figure 1: The constituent parts of an ontology

It is possible to draw a parallel with the world of relational databases, where the schema of a database can be seen as an ontology, and the data as instances or assertions that use the vocabulary of that ontology. However, there is a fundamental difference: a relational database assumes a closed world, which is not the case with ontologies, meaning that all information is present in the database, and anything that is not asserted is considered false.

Ontologies play an increasingly important role in knowledge management and are used as a standard representation of knowledge. Thanks to ontologies, users can understand each other by using a common understanding of a domain. This helps to understand the concepts of the domain, as well as enables the machine to interpret the definitions of concepts and their relationships. Ontologies primarily play a role in analyzing, modeling, and implementing domain knowledge, although they also influence knowledge related to problem-solving.

2.2 Legal Ontologies

The modeling and formalization of legal knowledge are crucial aspects to implement in order to improve legal assistance systems such as question-answering systems or legal information extraction systems. Considering that the basic types

of entities that populate the legal domain are assumed to be clearly identifiable and reasonably intersubjective, most efforts in the early 2000s focused on modeling foundational ontologies (high-level) and knowledge exchange formats that abstract legal denominators with a unifying vision of legal subdomains, enabling the reuse of ontologies and supporting their modeling in new legal subdomains. As a result, several ontologies have been proposed by researchers, whether foundational ontologies that define common concepts across all domains or core ontologies that define, for each relevant domain, a minimal set of generic and central concepts. Specifically, for the legal domain, the following can be found:

- ✓ DOLCE (Descriptive Ontology for Linguistic and Cognitive Engineering) [9][10]: is a foundational (high-level) ontology that provides a set of abstract concepts and relations to structure any domain, based on a fundamental distinction between enduring and perduring entities—two types of entities that do not exist in reality and have been subject to several criticisms. DOLCE+ is an extension of DOLCE that includes modules dedicated to core ontologies for time, space, plans, etc. I consider the taxonomy, especially concepts such as "enduring" and "perduring," less relevant for legal terms, as they do not represent any legal reality.
- ✓ LRI-Core [11][12]: is a core legal ontology that supports the development of ontologies in criminal law across various European countries within the e-Court project. It uses a different approach from other foundational ontologies: it does not distinguish between enduring and perduring entities, as in DOLCE, by considering all concepts as enduring (i.e., timeless) and all instances as perduring. Mental concepts are not treated as non-physical concepts, as in DOLCE, but rather the mental world is considered an analog of the physical world with an intentional perspective. The top of LRI-Core consists of five main categories: physical and mental concepts, roles, abstract concepts, and terms for events.
- ✓ CLO (Core Legal Ontology) [13]: extends DOLCE+, which is an extension of DOLCE, and defines legal concepts and relations based on its formal properties. CLO views the legal world as a description of social reality and relies on the distinction between descriptions, which encompass laws, norms, regulations, types of crimes, etc., and situations, which encompass facts and legal cases. It offers a more extensive classification of legal actions, including concepts such as legal facts, legal acts, intentional legal facts, etc. However, the classification of these actions as situations and the lack of distinction between individual actions and organizational actions remain subject to criticism.
- ✓ LKIF (Legal Knowledge Interchange Format) [14][15]: is a knowledge representation formalism that allows the translation of legal knowledge bases written in different representation formats and formalisms. It can be used as a central knowledge component for legal knowledge management systems. The LKIF core ontology consists of several modules, each describing a set of closely related concepts from legal and common-sense domains. For LKIF, the only concept that defines a legal action is the "act of law," defined as a public act by a legislative body that creates an action with legal status.

Despres and Szulman [16] provided a detailed comparison of core ontologies to help select an ontology suited to reuse constraints.

From the second decade of this century, efforts regarding the representation of legal knowledge shifted towards modeling specific legal subdomains, reflecting a growing awareness of the particularities that characterize them. These specificities led to a proliferation of ontologies and modeling vocabularies for different legal subdomains, making the reuse of legal knowledge more difficult, as it requires a broad understanding of the resources already available. As a result, several researchers found it valuable to conduct a state-of-the-art review and comparative analysis of existing legal ontologies and vocabularies. Casellas [17] proposed a comprehensive survey of legal ontologies covering a period of about fifteen years, from the 1990s to 2011. The characteristics of the ontologies she considered in her analysis mainly relate to the intended use of the ontology, its level of generality, its degree of formalization, the methodology used to build and evaluate the ontology, as well as its availability for reuse. De Oliveira Rodrigues et al. [18] extended the period of their literature review and analyzed legal ontologies proposed from the late 1990s to 2017. Their work presents various classification studies aimed at grouping ontologies based on different dimensions, some of which are similar to those already proposed by Casellas [17].

Leone et al. [19] focused their attention on recently published legal ontologies from the second

decade of this century, providing a state-of-the-art review and a practical information source to consult in order to make an informed and conscious decision about the knowledge already modeled and reusable from other ontologies. They analyzed a set of ontologies, which they grouped into five legal subdomains: Policies (ODRL [20], LDR [21]); Licenses (ccREL [22], L4LOD [23]); Tenders and procurements (LOTED2 [24], PPROC [25]); Privacy (Data Protection Ontology [26], GDPRtEXT [27], PrivOnto [28], PrOnto [29]); and Cross-domain Ontologies (Eurovoc [30], LegalRuleML [31][32], ELI [33], NRV [34]).

To model the criminal code targeted by this work, it is necessary to include concepts related to agents, actions, organizations, offenses, etc. As a result, several researchers have based their development of legal ontologies on foundational ontologies and core legal ontologies, which seem to be indispensable above an ontology of legal terms.

Dhouib and Gargouri [35] described the construction of an application ontology for the legal domain, specifically for the modules of legal actions and agents. They adopted a multi-layered approach with three levels of abstraction for the ontology's conceptualization: the most abstract level was based on the foundational (top-level) ontology DOLCE, which provides a set of abstract concepts and relations to structure any domain; the intermediate level was based on core legal ontologies CLO and LKIF, which define a minimal set of generic and central concepts for the legal domain; and finally, at the most specific level, the intermediate-level concepts were further refined by domain-specific concepts (e.g., sales contract, lawyer, decision, etc.). To conceptualize legal actions, which are actions carried out by actors in a legal context, they adopted the view that actions are perdurants controlled by at least one intention and distinguishable from events that lack intentional cause. Among the actions, deliberate actions, which are premeditated, were included. To conceptualize legal agents, who are actors capable of interacting in a legal context, they adopted the view that these are agentive entities that are endurants and encompass entities with the ability to carry out actions. They identified three types of legal agents: Legal Organizations, which represent agents performing legal acts, such as courts and tribunals; Professional Legal Agents, which represent agents in the legal profession, such as judges, lawyers, etc.; and Social Legal Agents, which represent agents who can interact in a legal context but are not part

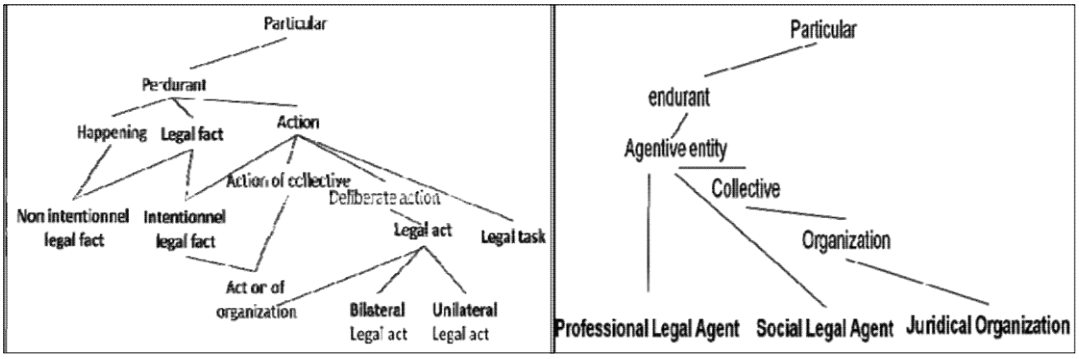
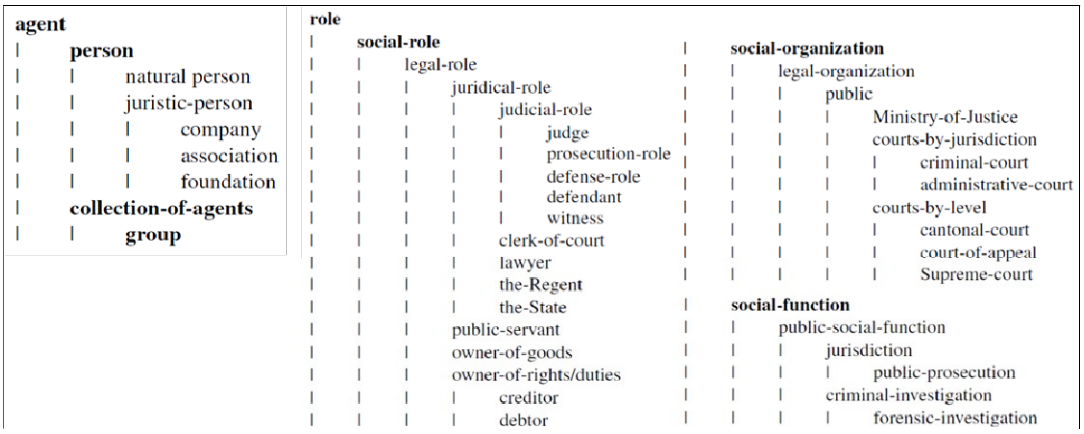


Figure 2: The Taxonomy Of Legal Agents And Actions Presented In [35]



Journal of Theoretical and Applied Information Technology
31st May 2025. Vol.103. No.10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

of the legal profession, such as witnesses, heirs, parties, etc. Figure 2 shows the taxonomy they presented for legal agents and actions.

Breuker et al. [36] developed an ontology, OCL.NL, that covers Dutch criminal law, based at the most abstract level on the concepts from the LRI-Core ontology. This Dutch ontology was intended to serve as a reference for developing similar ontologies for Italian and Polish criminal law. In OCL.NL, a distinction is made between agents as entities that act and the roles that an agent can fulfill. Agents include individuals, legal entities, and groups of people. Roles cover most social concepts: social organizations, where we find legal organizations such as ministries and courts; social roles, where we find legal roles such as judges and lawyers; and social functions, where we find jurisdictions. Figure 3 shows the taxonomy they presented for legal agents and roles.

Corcho et al. [37] constructed an ontology of legal entities within the context of Spanish law, based on a class taxonomy proposed by Breuker. They represented a taxonomy of legal entities by distinguishing between individuals and organizations. Individuals are further divided into natural persons, representing both adults and minors, and legal persons, representing companies (both public and private), associations, and foundations. Organizations represent ministries and courts. Figure 4 shows the taxonomy they presented for legal entities.

The consultation and analysis of these existing ontologies related to the area of interest provide an idea of the legal concepts already studied and help select relevant concepts for reuse. However, their organization and most of the concepts they define are not compatible with the specifics of the Moroccan Penal Code and do not cover its typical concepts very well. They also have strong common-sense appeal, but legal professionals, who are the primary target users, are mainly interested in the legal aspects as defined by the laws.

For the ontology by Dhouib and Gargouri [35], they used two types of offenses: the legal fact, which is an event likely to produce legal effects—either an intentional fact such as a murder or theft, or an unintentional fact such as a death or accident; and the legal act, which is the manifestation of will aimed at producing legal effects. They distinguish the legal fact from the legal act by intent, whereas in the Moroccan Penal Code, an offense is an act contrary to the law, which can be an act or an omission. However, I find the use of the concepts "legal organization," "professional legal person," and "social legal person" interesting.

For the ontology by Breuker et al. [36], I do not agree with the idea of introducing the role as a concept, as I consider it more of a property that a legal entity can have. However, I do agree with the idea of introducing an agent as a concept representing any entity that acts and is concerned with the law, but without specializing it into a "person" concept, as "person" could also represent a legal person who is rather supposed to enforce the law and is not, therefore, represented by "agent."

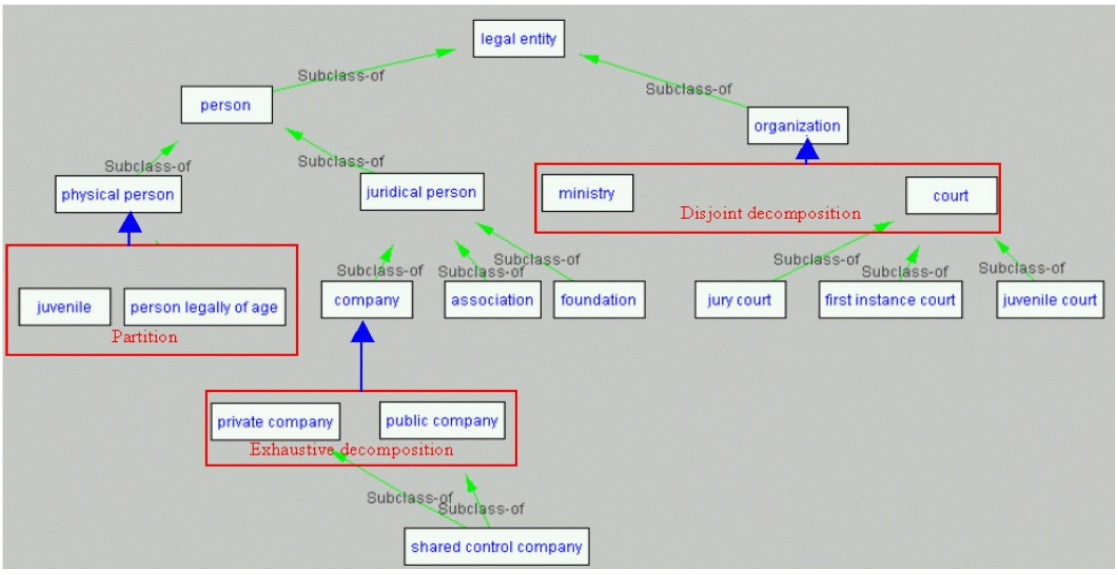


Figure 4: The taxonomy of legal entities presented in [37]

For the ontology by Corcho et al. [37], I do not agree with representing all types of people as legal entities, as I believe a distinction should be made between individuals who are supposed to enforce and uphold the law and individuals who are subject to the law, such as offenders, who I do not consider to be legal entities. Additionally, a specificity of the Moroccan Penal Code is that it distinguishes between minors under 12 years old and minors between 12 and 18 years old, who are treated and judged differently.

LKIF-Core presents the concept "legal_source," which represents most legal sources, such as legal documents, represented by the concept "legal_document," which is further specialized into several concepts {code, code_of_conduct, contract, decree, directive, regulation, statute, treaty}, representing most legal documents. CLO has the concept "legal document" without specialization. The grouping proposed in LKIF-Core better matches the reality and nature of the objects, which is why I selected this group of concepts as relevant for reuse.

2.3 Methodologies for Building Ontologies

The construction of an ontology for a particular domain requires an in-depth analysis to reveal the relevant concepts, attributes, relationships, constraints, instances, and axioms of that domain. Such knowledge analysis typically results in a taxonomy (is-a hierarchy) of concepts with their attributes, values, and relationships.

At the beginning of their emergence, the construction of ontologies was done in a rather ad hoc manner. In the meantime, several methodologies have been proposed to guide the ontology development process. I would like to mention six methodologies that I find the most representative, which have emerged to guide the ontology development process. Table 1 shows the most representative ontology construction methodologies along with the steps they define.

The methodologies that were presented with well-defined steps to guide developers in the ontology construction process are "METHONTOLOGY" and "Simple Knowledge-Engineering." A detailed example of the use of the "METHONTOLOGY" methodology was presented in the paper [43], where the authors demonstrated the use of METHONTOLOGY and ODE to construct an ontology of chemicals.

Unlike other methodologies, the "Simple Knowledge-Engineering" methodology [8], presented by Noy and McGuinness, who developed an initial ontology development guide, has a detailed implementation process that specifies several elements to guide an ontology developer. This methodology includes more steps that simplify the construction task for the developer. It includes the following elements:

1) Determine the domain and scope of the ontology: This step involves determining the domain, source, purpose, and scope of the ontology. The questions to address include: what domain will the ontology cover? What is the purpose of the ontology? What types of questions should the ontology help answer?

Table 1: The Most Representative Ontology Construction Methodologies

Year	Methodology	Ontology Development Process
1995	« TOVE » (Toronto Virtual Enterprise) [38]	(1) Identify motivating scenarios; (2) Define informal competency questions; (3) Define the ontology terminology; (4) Define formal competency questions; (5) Specify definitions and constraints on the terminology; (6) Test the competency of the ontology.
1996	« Skeletal Methodology » [39, 7]	(1) Identify the goal; (2) Build the ontology (Capture the ontology, Code the ontology, and Integrate existing ontologies); (3) Evaluation; (4) Documentation; and Guidelines for each phase.
1996	« knowledge conceptualization » [40]	(1) Capture knowledge; (2) Develop a requirements specification document; (3) Conceptualize the ontology; (4) Implement the conceptual model; and (5) Evaluation at each phase.
1997	« METHONTOLOGY » [41]	(1) Specification; (2) Knowledge acquisition; (3) Conceptualization; (4) Integration; (5) Implementation; (6) Evaluation; (7) Documentation. Knowledge acquisition, evaluation, and documentation are tasks carried out throughout the ontology lifecycle.
2001	« Knowledge Meta Process » [42]	(1) Feasibility study, (2) Ontology development launch, (3) Refinement, (4) Evaluation, (5) Maintenance.
2001	« Simple Knowledge-Engineering » [8]	(1) Determine the domain and scope of the ontology; (2) Consider the reuse of existing ontologies; (3) List the important terms of the ontology; (4) Define the classes and the class hierarchy; (5) Define the properties of the classes (slots); (6) Define the facets of the slots; (7) Create instances.

Who will use and maintain the ontology?

2) Consider reusing existing ontologies: This involves checking whether an ontology has already been developed in the same domain. If such an ontology exists, it is easier to modify the existing ontology to meet specific needs than to create a new one from scratch.
3) Enumerate important terms in the ontology: This step is considered the first step in the actual construction of the ontology. It involves creating a list of the expected terms that will be used in the construction of the ontology without worrying about overlaps between the concepts they represent or the relationships between them.
4) Define the classes and the class hierarchy: This step involves organizing the relevant terms identified hierarchically, using a top-down development approach, bottom-up development approach, or a hybrid (middle-out) approach that combines both top-down and bottom-up approaches.
5) Define the properties of class-slots: This involves identifying the properties (slots) of the classes in the remaining list of terms, indicating which class each property describes.
6) Define the facets of the slots: This step involves adding facets to the properties, which are restrictions on the properties, such as the type of value, allowed values, number of values (cardinality), and other characteristics of the values the slot can take.
7) Create instances: This step involves creating instances of the classes, referred to as individuals, and filling in the slot values.

This final "Simple Knowledge-Engineering" methodology has been widely used by several researchers who adapted it to their needs. Boyce and Pahl [44] used this methodology with some adaptations for constructing an ontology of course subjects. Alfaifi [45] demonstrated the use of this methodology for the development of an ontology for information technologies at the University of Tabuk.

Despite the differences between methodologies in the order of steps and the language used to define the steps, they all adopt conceptually similar steps and requirements, which can be defined as: specification, conceptualization, implementation, evaluation, and documentation. Conceptualization is the main operation, which involves identifying terms, grouping them into semantic classes, and structuring them into a terminological network. However, all these methodologies summarize ontology development as a mere enumeration of important terms, followed by the definition of classes and their hierarchy, the specification of properties and their facets, and finally the creation of instances—without showing users the approach or method that could guide them in selecting terms, defining classes, hierarchy, and properties, or in showing them how to build multilingual ontologies. In addition, there is a lack of models to represent ontology data in a way that supports its development and documentation.

2.4 Non-manual Methods for Ontology Construction

The construction of an ontology is not a simple task. It requires time, effort, and expertise in the domain in which we wish to build the ontology. Normally, a team of individuals, such as domain experts and ontology engineers, are responsible for the development of the ontology. As a result, researchers have turned to non-manual methods for constructing ontologies from texts. The non-manual construction of ontologies from texts is a subfield of ontology engineering in its own right. The use of texts is justified by linguistic research, whose main hypothesis is that texts carry stabilized knowledge shared by communities. Moreover, even though they do not completely replace experts, texts are more readily available than experts, who often lack the time to participate in the construction process.

A four-step methodological framework is common to most non-manual methods for constructing ontologies from texts: 1) constructing a document corpus; 2) linguistic and statistical analysis of the corpus; 3) conceptualization; and 4) operationalizing the ontology. These relatively independent steps perform a dual movement, transitioning from informal to formal, moving from the textual level where knowledge is described in corpora to the conceptual level where knowledge is described through concepts denoted by linguistic entities and the relationships between these concepts.

In the preparatory phase of conceptualization, three main operations can be distinguished: Identifying terms; Grouping terms into semantic classes; and Structuring the classes into a terminological network. There are three types of ontology construction systems where the conceptualization is carried out automatically, semi-automatically, or manually assisted [46]:

- ✓ Automatic Ontology Construction Systems: These systems enable the fully automatic construction of ontologies from texts, such as

Text2Onto [47], which extracts concepts, relationships between these concepts (equivalence relations, hierarchical relations, etc.), and instances of concepts from texts.

- ✓ Semi-automatic Ontology Construction Systems: These systems allow for interactive ontology construction from texts, such as OntoGen [48], which suggests concepts to the domain expert in the form of document classes, proposes a designation, and automatically associates instances (documents) with them.
- ✓ Assisted Manual Ontology Construction Systems: These systems support interactive ontology construction from texts, such as Terminae [49], which guides the ontologist through the ontology design process.

In general, methods for identifying terms and relationships between them from texts rely on natural language processing (NLP) techniques, which use linguistic methods, statistical methods, or a combination of both (hybrid):

- ✓ Linguistic methods: These involve performing a syntactic analysis on texts, identifying nouns, verbs, adjectives, and adverbs, as well as the syntactic dependencies between them (subject of the verb, object of the verb, etc.). By applying a set of syntactic rules, it is possible to identify complex terms such as noun phrases, verb phrases, adjective phrases, etc. There are several tools available for constructing and applying linguistic rules to texts, such as NooJ [50].
- ✓ Statistical methods: These involve performing statistical calculations using statistical measures to identify terms, such as:
  - tf (Term Frequency): Refers to the number of times a given term appears in the corpus.
  - idf (Inverse Document Frequency): Establishes the distribution of terms within a corpus, based on the principle that the importance of a term is inversely proportional to the number of documents in the corpus in which the given term appears.
  - tf-idf (Term Frequency-Inverse Document Frequency): Combines tf and idf with the idea of distinguishing terms that, although appearing in a small number of documents in the corpus, also have a high-frequency rate within the corpus.
  - Entropy: Used to measure disorder, based on the ratio between the frequency of a term in a document and the total frequency of the term in the corpus.

Lame [51] presented a method that relies on natural language processing (NLP) techniques by combining syntactic analysis and statistical analysis to extract concepts and the relationships between them to build a legal domain ontology dedicated to information retrieval. The author used the "Syntex" parser, which allows for syntactic analysis of texts by identifying nouns, verbs, adjectives, adverbs, and syntactic dependencies between them (subject of the verb, object of the verb, etc.). He then applied statistical methods to identify in the list of extracted terms those that could be classified as legal terms and those that could not. He only considered terms belonging to a single syntactic category: nouns and noun phrases, assuming that most concepts are encapsulated in nouns. He also removed terms containing non-alphabetic characters from the initial list, considering that numbers are not critical when the ontology is dedicated to information retrieval rather than reasoning. Then, he used statistical methods such as tf, idf, tf-idf, and entropy to weight the terms and determine which ones are legal and which are not. However, the results he obtained did not allow for the identification of legal terms without manual intervention.

In conclusion, despite advancements in the field of natural language processing, their use remains unsatisfactory because they allow for the identification of certain terms without enabling a strict identification of all terms within the study domain, while still requiring manual intervention to validate the relevance of the identified terms. This can be explained by the fact that, on one hand, linguistic methods, while capable of developing rules to identify domain terms, can also identify terms outside the domain due to the lack of specific linguistic structures for domain terms. On the other hand, statistical methods can identify terms that are unrelated to the domain despite their high or low frequencies. As a result, the various methods proposed and developed for term extraction and the identification of conceptual vocabulary from texts cannot be fully automated, as the results from extractors are noisy, and terminological judgment is partly subjective, especially in complex domains like the legal field.

2.5 Ontology Modeling Languages

To build an effective ontology, a modeling language for ontologies is used to describe explicit and formal conceptualizations of a given domain.

Journal of Theoretical and Applied Information Technology
31st May 2025. Vol.103. No.10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Ontology modeling languages are formal languages that allow for the formal description of concepts, relationships between concepts, properties, property characteristics, and instances.

The main impetus for ontologies came from the vision of the Semantic Web, as formulated by Tim Berners-Lee (director of W3C and inventor of the Web, HTTP, and HTML) [52]. The idea behind the Semantic Web is to make information understandable by machines. It represents an extension of the World Wide Web in which information is provided with a well-defined meaning, transforming unstructured data into semi-structured data in a form understandable by machines. Ontologies have contributed to the development of the Semantic Web by providing the conceptual foundation that allows the semantics of metadata to be interpretable by machines. These ontologies represent information in a way that can be used by machines, not only for display but also for automation, integration, and reuse across various applications.

In 2002, the W3C, as part of its work on the Semantic Web, established a working group dedicated to developing standard languages for modeling data and ontologies. This group led to the emergence of technologies and languages that revolutionized the way we use the Web, and more generally, how we represent semantic data to find and organize information. These technologies and languages are combined to provide descriptions that complement or replace the content of documents. Thus, the content can be manifested as descriptive data stored in databases accessible on the Web or through tags in documents, via HTML or its variant XHTML. The languages and technologies developed have been defined in the Semantic Web Stack, which illustrates the hierarchy of computer languages [53]. This pyramid is continuously evolving. Figure 5 illustrates the Semantic Web Stack in its latest version [54].

At the bottom of the architecture, we find URIs (Uniform Resource Identifiers) and IRIs (Internationalized Resource Identifiers) [55]. A URI extends the concept of a URL, where a URI identifies a unique object (such as a person, place, etc.) and a URL identifies a web page designed to be displayed in a browser. An IRI is an extension of the URI concept that allows the use of non-ASCII characters, including characters from non-Latin alphabets such as Arabic, Chinese, or Cyrillic.

XML (eXtensible Markup Language) is a generic markup computer metalanguage that derives from SGML [56]. It allows the description of structured documents according to predefined directives or syntax. XML is particularly well-suited for sending documents across applications.

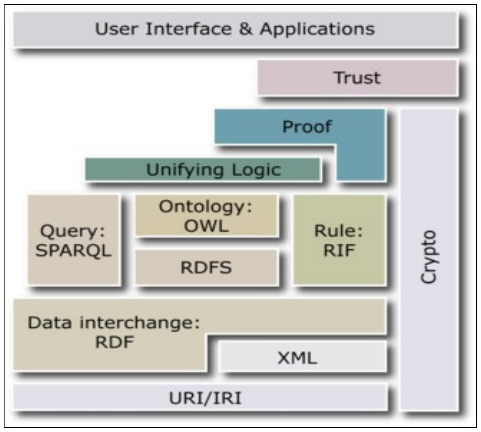


Figure 5: The Semantic Web Architecture

RDF (Resource Description Framework) [57][58][59] is a framework that allows representing information resources (people, places, animals, documents, concepts, etc.) in a machine-understandable way, in the form of a "subject, predicate, object" triple where all elements of the triple are resources, except for the last element, the object, which can also be a literal, such as a constant value like a string or a number. It identifies objects using Web identifiers (IRI) and describes resources with properties and property values. An RDF-based model can be represented through various data exchange syntaxes for communication between different applications, such as RDF/XML, RDF/JSON, N3, Turtle, N-Triples, RDFa, Microformat, and Microdata. RDF triples are visualized as directed and labeled graphs in which subjects and objects are represented by nodes and predicates by arcs (see Fig. 6.).

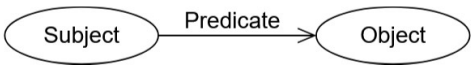


Figure 6: The representation of RDF triples as a directed and labeled graph

RDFS or RDF Schema (Resource Description Framework Schema) is an extension of RDF that provides a vocabulary for modeling RDF data [60][59]. It does not provide specific classes and properties for applications, but rather provides modeling primitives to describe application-specific classes and properties, and organize information into hierarchies. Its key primitives are classes and properties, subclass and subproperty relationships,

as well as domain and range restrictions (rdfs:Class, rdfs:subClassOf, rdfs:subPropertyOf, rdfs:domain, or rdfs:range). It is one of the first languages for formalizing and describing ontologies, as well as producing knowledge representation documents. This allows resources to be defined as instances of classes and subclasses of classes.

OWL (Web Ontology Language) is a language based on the RDFS standard [61][62]. More expressive than its predecessor RDFS, OWL quickly became dominant in the field of ontologies and is now the most widely used standard. As with any language or database, there is a lower level that describes how concepts are mapped to real-world data. In a relational database, the fundamental structure for representing data is a table. In OWL, the fundamental structure is a triple. It is a language that provides a formal description of concepts, terms, or relationships of any given domain to describe the semantics of data through ontologies. It adds more vocabulary to describe properties and classes, such as relationships between classes, cardinality, equality, property typing, and property characteristics such as symmetry, etc. The OWL language is available in several levels: OWL-Lite, OWL-DL, and OWL-FULL, which are nested within one another.

SPARQL is a query language just like SQL for relational databases. Essentially, SPARQL is to the Semantic Web and knowledge graphs what SQL is to relational databases. Just like SQL, SPARQL allows for querying, but on RDF triples and graphs, enabling the search, addition, modification, or deletion of RDF data [63][64].

An excellent book on the SPARQL language is “Learning SPARQL” by Bob DuCharme, published by O'Reilly [65]. This book not only covers SPARQL but also topics such as RDF/RDFS and how triples are used to represent information in OWL.

2.6 Ontology Editors

Editing an ontology with the appropriate tool allows it to be displayed in a tree structure. Furthermore, the integration of suitable plugins enables the visualization of various concepts and all the relationships connecting them, providing a more comprehensive view of how the concepts are arranged to one another. Some editors allow importing or exporting an ontology from one format to another, which greatly facilitates its portability and the automatic generation of OWL/XML or RDF files.

There are various tools available for ontology development, such as Protégé, DOME, OntoLingua, and Altova SemanticWorks. Protégé is one of the most widely used ontology editors, distributed as open source by the Stanford Medical Informatics Institute. It supports several ontology representation languages, including OWL. An ontology in Protégé can be exported to various formats, including RDF(s), OWL, and XML schemas. Protégé is a Java-based platform, which is flexible and supports multiple languages, including English, French, Arabic, Chinese, Russian, and more. In addition to visualizing the ontology hierarchy, Protégé allows graphical visualization through plugins such as OntoGraph or OWL-Viz. It also has reasoners like Racer, Fact++, Hermitt, and Pellet. OntoGraph, Fact++, Hermitt, and Pellet are bundled with Protégé. A large community of academic developers, governments, and companies use Protégé in various fields.

Ontology development using Protégé has been illustrated by some researchers, such as Jain and Singh [66], who highlighted the explicit description of the ontology concept and the development and methodology involved in building the ontology using Protégé; and Alfaifi [45], who described the process of developing an ontology with Protégé for the Department of Information Technology at the University of Tabuk.

3. TO-MULTILONTOLOGY & MPCO: A METHODOLOGY FOR DEVELOPING MULTILINGUAL ONTOLOGIES & A LEGAL ONTOLOGY OF THE PENAL CODE

Ontologies are based on the RDF model, which allows knowledge to be represented and exchanged in the form of triples: (subject, predicate, object). For example, to model in RDF the statement "L’attentat contre la vie du roi est puni de mort" [The attempt against the life of the king is punishable by death], we break it down into controlled language into elementary units, each of which can be represented by a triple:

- ✓ "L’attentat" contre la vie du "roi" ["The attempt" against the life of the "king"]
- ✓ "L’attentat" est puni de "mort" ["The attempt" is punishable by "death"]

This results in two triples, both of which have "attentat" [attempt] as the subject:

- ✓ (attentat, contre la vie de, roi) [(attempt, against the life of, king)]

Journal of Theoretical and Applied Information Technology
31st May 2025. Vol.103. No.10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

✓ (attentat, est puni de, mort) [(attempt, is punishable by, death)]

These ontological knowledge representations are often shown in the form of trees because there is a taxonomy of classes and categories we use (see Fig. 7).

Figure 7: Tree Representation Of Ontological Knowledge

Now, let's differentiate between what we call a concept and a term. Indeed, there are several ways to refer to the term "death." For example, we can say "mort" in French, "death" in English, and "إعدام" in Arabic. However, these three terms can refer to the same notion, the same concept, which is shared among English, French, and Arabic speakers. So, we will say here, for instance, that there exists a concept—let's call it concept #B—that represents the term "death," and we will associate with this concept the term (label or tag) "mort" in French, "death" in English, and "إعدام" in Arabic. Similarly, there is a concept—let's call it concept #A—that represents "assault," and we will associate with this concept the term "attentat" in French, "attack" in English, and "اعتداء" in Arabic (see Fig. 8). Likewise, properties can be represented by terms. These identifiers, their organization, and the logical dependencies between them are what we call a formal ontology. It is this domain-specific legal knowledge that we will encode in a machine so that it can simulate the inferences we make naturally.

In practical terms, the development of an ontology involves the following steps:

- ✓ Defining the classes of the ontology;
- ✓ Organizing the classes into a taxonomic hierarchy (subclass, superclass);
- ✓ Defining the slots and the descriptions of the allowed values for these slots, as well as the restrictions on the slots, known as facets;
- ✓ Defining the instances, also called individuals;
- ✓ Populating the values for the slots of the instances.

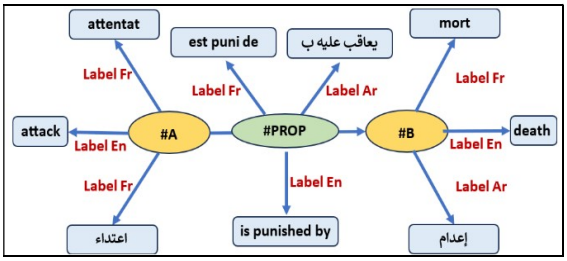


Figure 8: Graph Representing The Concepts Of The Ontology And Their Labels

Defining the instances of the classes by filling in the values for the slots creates a knowledge base.

An ontology for a large and complex domain can be built from smaller ontologies by breaking down the domain knowledge into smaller pieces. Given that the legal domain is vast and complex, I opted for a modular ontology based on code, starting with the Moroccan government's penal code.

After a thorough review of all the methodologies described in the state-of-the-art ontologies, I chose to adopt a customized methodology, TO-MULTILONTOLOGY, which covers aspects ranging from the specification phase to the validation and evaluation phase, with a detailed implementation process that outlines clear steps to guide and simplify the task of constructing multilingual ontologies. This solution also addresses one of the main barriers to the effective sharing of knowledge, which is the inadequate documentation of existing ontologies, by providing powerful tools and models that allow both documenting the ontology and guiding its development.

This methodology will be explained and used in the development of a multilingual legal ontology, MPCO (Multilingual Penal Code Ontology), in French and Arabic, for the Moroccan government's penal code. The constructed ontology can play an important role in information retrieval and in learning the knowledge of the penal code and can also serve as a reference for the development of similar penal law ontologies.

The TO-MULTILONTOLOGY methodology proposed for the development of multilingual ontologies is outlined in Figure 9. It consists of several activities that will be described in detail in the following subsections:

1) Establish the ontology charter - Specification;
2) Build the ontology skeleton;

Journal of Theoretical and Applied Information Technology
31st May 2025. Vol.103. No.10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3) Define the basic properties between the core concepts of the ontology skeleton;
4) Conceptualize and refine the ontology;
5) Identify and create the individuals;
6) Verify the consistency of the ontology and simulate deductive reasoning;
7) Validate and evaluate the ontology.

The tool used for the construction and development of the ontology is Protégé 5.6.4, which is a free and open-source tool for editing and managing ontologies. Michael DeBillis created a detailed guide on using Protégé in its version 5.5 for ontology development [1] that I recommend reading before following this ontology development guide.

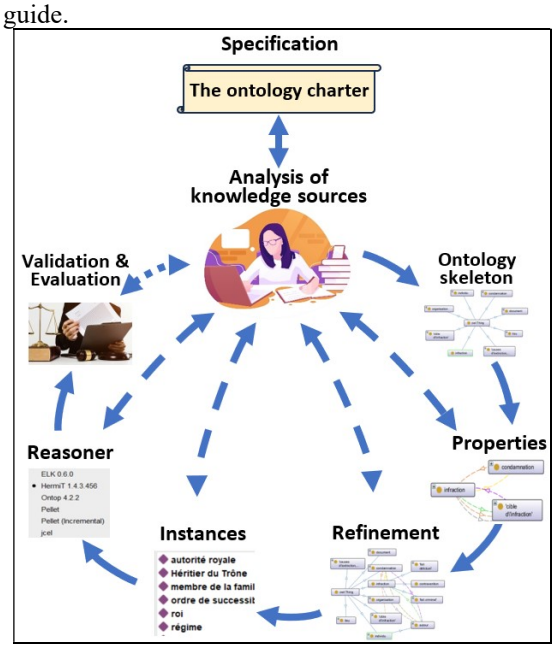


Figure 9: The Process Of The TO-MULTILONTOLOGY Methodology

The tool used for the construction and development of the ontology is Protégé 5.6.4, which is a free and open-source tool for editing and managing ontologies. Michael DeBillis created a detailed guide on using Protégé in its version 5.5 for ontology development [1] that I recommend reading before following this ontology development guide.

3.1 Establish the ontology charter - Specification

For ontology development, as in any project, the first challenge to tackle before even starting the development of an ontology is to frame its development and prepare for it. When one is not an expert in the domain, the first thing to do is to acquire a minimum level of knowledge about the domain based on knowledge sources. Experts, books, manuals, figures, tables, and even other ontologies are knowledge sources from which insights can be derived. It is also useful, before starting the development of an ontology, to review other ontologies that have been published and are related to the domain of interest.

Thus, I acquired knowledge about the domain by conducting a preliminary study of the penal code to develop a preliminary version of the required specification. Additionally, several ontologies in the legal domain were consulted, particularly those related to the penal code, to understand the legal concepts that have already been studied. These ontologies are described in the subsection on legal ontologies. However, their organization and most of the concepts they define are not compatible with the specificities of the Moroccan penal code and do not cover its typical concepts very well. They also carry a strong flavor of common sense, but legal professionals, who are the primary users, are primarily concerned with the legal aspects as defined by the laws.

As a result of this preliminary study, I drafted a document that can be called the ontology charter, the equivalent of a project charter for project development, which constitutes the best way to have a global vision before starting. This charter represents the birth certificate of the ontology as it provides all the necessary information for the ontology. Figure 10 shows the model of the ontology charter, which outlines the requirements for the multilingual ontology of the legal domain, particularly the Moroccan government penal code.

This charter consists of four main parts:

1) The first part describes the ontology by specifying: the domain of the ontology and its objective, the target applications of the ontology, the knowledge sources used for the ontology's design, the languages and tools used for its development, the scope of the ontology, and the languages supported by the ontology;
2) The second part specifies the authors who contributed to the development of the ontology, the role of each, and their contact information;
3) The third part is dedicated to the management of versions and successive modifications of the ontology;
4) Finally, the last part formalizes the validation and evaluation of the ontology. This validation is carried out with domain experts as well as through the use of competency questions.

Competency questions are defined as a set of questions expressed in natural language that the constructed ontology must be able to answer correctly [38][67].

I have constructed a set of natural language questions that will be considered as requirements to which the ontology must be able to respond, to evaluate and guide the ontology construction process. These are informal competency questions that will ensure that the ontology can respond correctly to the requirements and provide an initial evaluation of the ontology. They are informal because they have not yet been expressed in the formal language of the ontology, which is SPARQL.

Charte d'ontologie

Description d'ontologie

Domaine	Juridique (Code pénal du Maroc)
Objectif	Ontologie multilangue qui stricture le code pénal du gouvernement marocain
Applications	Systèmes de question-réponse
Sources de connaissances	Code pénal du gouvernement marocain
Langages	RDF, RDFS, OWL
Outils	Protégé 5.6.4
Portée	Les infractions, les peines, les victimes, les cibles, les juridictions, ...
Langues	Français et Arabe avec possibilité d'ajouter d'autres langues

Auteurs

Nom / mél	Qualité / rôle
Ismahane KOURTIN / kourtin_ismahane.math@yahoo.fr	Responsable du projet

Historique des modifications et révisions de ce document

n° de version	Date	Description et circonstances de la modification
V 0.1	01/03/2024	Première version à la suite d'une première étude du code pénal
V 1	01/06/2024	Compléments d'informations
V 1.1	16/12/2024	Validation

Validations

n° de version	Nom / qualité	Date / signature	Commentaires et réserves éventuelles
V 1.1	Juriste		

N°	Questions de compétence informelles	Questions de compétence formelles	Validation
Q1	Quels sont les peines qui réprime une infraction ?		
Q2	Quels sont les types d'infractions couverts par le code pénal ?		

Figure 10: The Ontology Charter

These competency questions were chosen to cover all aspects of the penal code. They will be used to evaluate the ontological commitments made and the expressiveness required of the ontology to represent questions related to the penal code.

These questions cover most of the topics addressed in the penal code, such as:

- ✓ The architecture, categories, and types of a given entity, such as searching for the types or categories of penalties or offenses: What are the categories of penalties?
- ✓ Entities with a property that has a value, such as searching for the penalties that punish a given offense: What are the penalties for espionage?
- ✓ Entities with a property that has a type of value, such as searching for the penalties that punish offenses of treason: What are the penalties for an act of treason?
- ✓ Entities with a property that has a value, which also has a property with a value, such as searching for the penalties that punish an attack against the king's person: What are the penalties for an attack against the king's person?
- ✓ Entities with a property that has a value, which also has two properties with values, such as searching for the penalties that punish an attack against the king's person that caused results: What are the penalties for an attack against the king's person that resulted in harm?
- ✓ The properties of an entity that has a property with a value, such as searching for cases where an attack is punished by death, when possible penalties for an attack differ, such as death and life imprisonment: In which cases is an attack punished by death?
- ✓ The place of execution of a penalty, such as searching for the place of execution of life imprisonment: Where is life imprisonment carried out?
- ✓ Offenses that are punished by a given sentence, such as searching for offenses that are punishable by death: What are the offenses punishable by death?

Once the competency questions are posed informally and the ontology's terminology is defined, they are translated into formal questions known as SPARQL queries, using the SPARQL language, which allows for querying ontologies and RDF graphs.

3.2 Build the ontology skeleton

After acquiring the necessary knowledge by studying the knowledge sources of the domain of interest (the penal code in this case), and after developing a preliminary version of the ontology charter that defines the scope of the ontology, the identification of core concepts, or "generic legal entities," is carried out. These are the main concepts that pertain to the domain of interest. This is the first step in constructing the ontology and specifying its terminology, which involves identifying the objects within the discourse domain.

Other concepts will be identified as the ontology-building process progresses.

To identify the core concepts related to the penal code, it is necessary to learn as much as possible about the domain of expertise (the penal code), which can be achieved by studying and analyzing the documentation. The initial study and analysis of the documentation involves identifying and extracting the core concepts covered by the penal code.

It is important to keep in mind that knowledge acquisition is an independent activity in the ontology development process. Most of the acquisition occurs during the requirements specification phase, and it decreases as the ontology development progresses.

The extraction of core concepts should be accompanied by what can be called the "core concepts dictionary," which helps fill in information about these concepts, such as the concept code and a label for each language targeted by the ontology, in this case, French and Arabic. Additional properties can also be added, such as the description of the concept. For my part, I used an Excel spreadsheet for the entire ontology, where I represented the dictionaries through Excel sheets. I named the core concepts dictionary sheet "CONCEPTS_NOYAUX."

Given that a legal system aims to regulate a society, it is important to represent the members of that society. I consider that a society is composed of agents and legal entities. Agents can be individuals, organizations, or groups of individuals who must adhere to a code of conduct. Legal entities are those that are supposed to represent the law and can be legal organizations, professional legal persons, or social legal persons. As for the legal sources that regulate society, they are represented by the concept of "legal source," which is defined in the core ontology LKIF-Core. These three concepts, "agent," "legal entity," and "legal source," can be reused in other ontologies for other legal sub-domains.

I have identified nine core concepts covered by the penal code, namely:

- ✓ "agent" [agent]: Represents entities that must adhere to a code of conduct and can be perpetrators of offenses. These include individuals, organizations, or groups of individuals.
- ✓ "entité juridique" [legal entity]: Represents entities that are supposed to represent the law and may include legal organizations, professional legal persons, and social legal persons.
- ✓ "infraction" [offense]: Represents human actions that, due to the social disturbance they cause, are considered offenses.
- ✓ "condamnation" [conviction]: Represents the judicial decision in which an agent is found guilty of an offense and is subject to a criminal sanction.
- ✓ "cause d'arrêt des condamnations" [cause of cessation of convictions]: Represents the causes for the extinction, exemption, and suspension of convictions.
- ✓ "lieu de condamnation" [place of conviction]: Represents the locations where convictions are carried out.
- ✓ "cible d'infraction" [offense target]: Represents the entities that can be victims of an offense, whose nature determines the type of convictions.
- ✓ "source juridique" [legal source]: Represents all legal sources such as legal documents like the penal code, directives, decrees, etc.
- ✓ "action juridique" [legal action]: Represents legal actions such as trials.

Table 2 illustrates the dictionary of core concepts, showing the main classes of the penal code ontology and their properties.

Journal of Theoretical and Applied Information Technology
31st May 2025. Vol.103. No.10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Once the editing of the core concepts dictionary is completed, we proceed to the creation of the ontology skeleton by implementing the dictionary data using the Protégé tool. The concepts from the dictionary will become classes in the ontology and serve as anchor points in the class hierarchy. For each concept, we define a class with an IRI carrying the concept code, and then we add the properties of the concept (labels and descriptions) as annotation properties. Labels are represented by the annotation property rdfs:label, which allows us to attach a label to the concept, specifying the corresponding language for the annotation. We then add a label for the concept's name in French and a label for its name in Arabic. Multiple labels in the same language can also be added if the concept has several terms in the same language. Descriptions are represented by the annotation property rdfs:comment, which allows us to attach a comment to the concept, specifying the corresponding language for the annotation. We add a comment for the concept description in French and a comment for its description in Arabic. If the available annotation properties are insufficient to represent the concepts' properties, we add others.

The knowledge thus acquired, expressed in natural language, is encoded and stored in the ontology by creating the first basic architecture that represents the skeleton of the ontology. Figure 11 shows the tree visualization of the ontology skeleton hierarchy in French on the left and in Arabic on the right, and Figure 12 shows its graphical visualization in French on the left and in Arabic on the right.

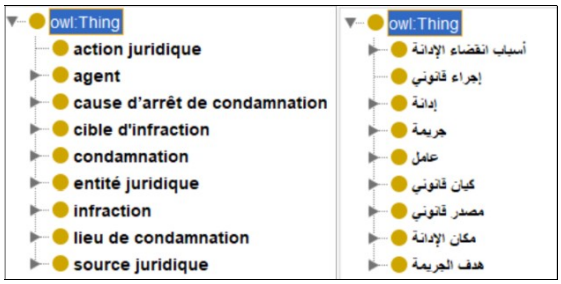


Figure 11: The tree visualization of the ontology skeleton

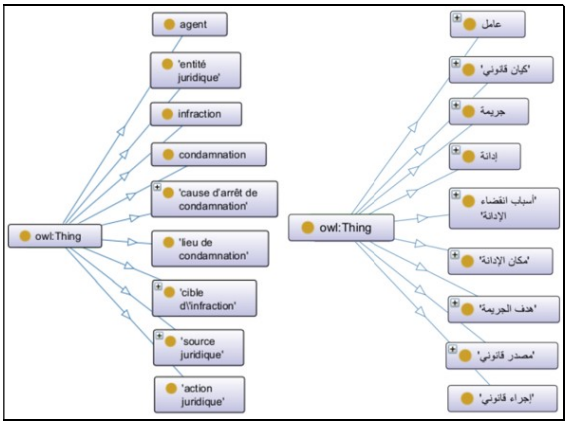


Figure 12: The graphical visualization of the ontology skeleton

3.3 Define the basic properties between the core concepts of the ontology skeleton

A crucial step in ontology construction is properly defining the properties (relations) that link the concepts in the ontology. This step essentially involves defining the properties between the core concepts of the ontology skeleton, which were identified in the previous step.

There are three types of properties:

Table 2: Dictionary of Core Concepts of the Ontology

Concept code	Label Fr	Label Ar	Label En
AGENT	agent	عامل	agent
ENTITE_JURIDIQUE	entité juridique	كيان قانوني	legal entity
INFRACTION	infraction	جريمة	offense
CONDAMNATION	condamnation	إدانة / حكم	conviction
CAUSE_ARRET_CONDAMNATION	cause d'arrêt de condamnation	أسباب انقضاء الإدانة	cause of termination of conviction
LIEU_CONDAMNATION	lieu de condamnation	مكان الإدانة	place of conviction
CIBLE_INFRACTION	cible d'infraction	هدف الجريمة	target of offense
SOURCES_JURIDIQUE	source juridique	مصدر قانوني	legal source
ACTION_JURIDIQUE	action juridique	إجراء قانوني	legal action

- ✓ Annotation properties: These are relations between an entity and a value of a given type, such as a string ("xsd:string") or an integer ("xsd:integer"). They typically represent metadata, such as "rdfs:label" and "rdfs:comment," which link string-type values ("xsd:string") to an entity.
- ✓ Object properties: These are relations between two individuals.
- ✓ Data properties: These are relations between an individual and a value of a given type, such as a string ("xsd:string") or an integer ("xsd:integer").

In OWL, only individuals can have values for object and data properties, but any entity can have a value for an annotation property since metadata applies to all entities. Annotation properties generally cannot be used for reasoning and can be assigned to classes, individuals, or even properties. This is the case for annotation properties like "rdfs:label," which assigns a label (xsd:string value) to an entity, and "rdfs:comment," which assigns a comment (xsd:string value) to an entity. These are used to assign linguistic data to the objects in the ontology, whether they are classes, individuals, or properties.

Identifying object properties that represent the relationships between core concepts and their restrictions can be done through the analysis of textual structures that link the concepts. For example, for "condamnations" [sentences] and "infractions" [offenses], among the structures that have been identified are: the structure "des peines prévues pour réprimer les faits constituant des infractions" [penalties provided to repress actions constituting offenses] which allows identifying the "réprime" [represses] object property linking "condamnations" [sentences] and "infractions" [offenses], and which is restricted to the domain (subject type) "condamnation" [sentence] and the range (object type) "infraction" [offense]; and the structure "l'attentat contre la vie ou la personne du Roi est puni de mort" [an attack on the life or person of the King is punishable by death] which allows identifying the "est puni de" [is punished by] object property linking "infractions" [offenses] and "condamnations" [sentences], and which is restricted to the domain "infraction" [offense] and the range "condamnation" [sentence]. Additional restrictions and characteristics can also be added to properties, such as "inverse property," "transitivity," "reflexivity," etc. In the previous example, the "réprime" [represses] property is the inverse of the "est puni de" [is punished by] property. These restrictions aim to limit the operations of a class or a property. Figure 13 shows a simplified conceptual graph of the two concepts "infraction" [offense] and "condamnation" [sentence] and their relationships. The linguistic data for classes and properties are described using annotation properties.

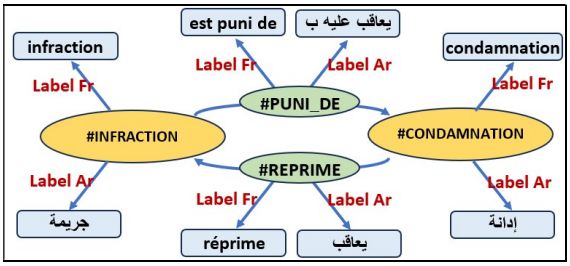


Figure 13: The simplified conceptual graph of the two concepts "infraction" [offense] and "condamnation" [sentence] and their relationships

Journal of Theoretical and Applied Information Technology
31st May 2025. Vol.103. No.10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

This study must be accompanied by what can be referred to as the object properties dictionary, which allows for recording information about the properties, such as: the property code, a label for each language targeted by the ontology (in this case, French and Arabic), the domain, the range, the inverse property, transitivity, reflexivity, etc. Linguistic variations (synonyms and acronyms) representing a property in the same language can also be added. I included an Excel sheet in the ontology workbook for the object properties dictionary, which I named "PROPRIETES_OBJETS." I started with 14 object properties that link the core concepts of the ontology's skeleton, representing relationships between them. Table 3 shows an excerpt from the ontology's object properties dictionary.

For example, the class #INFRACTION (offense) is linked to the class #CONDAMNATION (sentence) through the object property #PUNI_DE (is punished by). This object property #PUNI_DE has multiple labels, represented using the annotation property "rdfs:label." In French, the labels capture linguistic variations of this property: "est puni de" and "est réprimé par." In Arabic, the labels reflect linguistic variations in that language: "يعاقب ب" and " يعاقب عليه ب."

Once the editing of the relations dictionary is complete, the properties and their restrictions are implemented by adding them to the ontology skeleton using Protégé. Figure 14 shows the hierarchical visualization of a subset of the ontology's object properties.

Figure 14: The hierarchical visualization of a subset of the ontology's object properties

Table 3: The ontology's object properties dictionary

Property code	Label Fr	Label Ar	Domain	Range	Inverse property
CONTRE	contre	ضد	INFRACTION	CIBLE_INFRACTION	VISE_PAR
VISE_PAR	est visé par	استهدف ب	CIBLE_INFRACTION	INFRACTION	CONTRE
PUNI_DE	est puni de est réprimé par	يعاقب ب يعاقب عليه ب	INFRACTION	CONDAMNATION	REPRIME
REPRIME	réprime est édicté pour est prononcé pour	يعاقب مقرر ل يحكم به ل	CONDAMNATION	INFRACTION	PUNI_DE
EDICTEE_PAR	est édicté par est établi par	مقرر من قبل	CONDAMNATION	ENTITE_JURIDIQUE	PRONONCE
PRONONCE	prononce édicte ordonne prescrit	يقضي ينص على يحكم يحدد	ENTITE_JURIDIQUE	CONDAMNATION	EDICTEE_PAR
APPLIQUEE_A	est appliqué à	تطبق على	CONDAMNATION	AGENT	CONDAMNEE_A
CONDAMNEE_A	est condamné à est soumis à	يعاقب ب	AGENT	CONDAMNATION	APPLIQUEE_A
COMMET	commet réalise	يرتكب	AGENT	INFRACTION	COMMISE_PAR
COMMISE_PAR	est commise par	يرتكبها	INFRACTION	AGENT	COMMET
SEXECUTE_DANS	s'exécute dans	تنفذ داخل	CONDAMNATION	LIEU_CONDAMNATION	OU_SEXECUTE
OU_SEXECUTE	où s'exécute	اين تنفذ	LIEU_CONDAMNATION	CONDAMNATION	SEXECUTE_DANS
CAUSER_ARRET	cause l'arrêt de arrête	يوقف يتسبب في انقضاء	CAUSE_ARRET_CONDAMNATION	CONDAMNATION	EST_ARRETE
EST_ARRETE	est arrêté par	يتم إيقافه ب	CONDAMNATION	CAUSE_ARRET_CONDAMNATION	CAUSER_ARRET

Journal of Theoretical and Applied Information Technology
31st May 2025. Vol.103. No.10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3.4 Conceptualize and refine the ontology

This step involves conceptualizing the knowledge acquired from informal and unstructured knowledge sources by organizing and structuring it into a hierarchical taxonomy. Before proceeding with conceptualization, dictionaries of specific concepts are created, similar to the dictionary of core concepts. For each core concept, a dictionary of specific concepts is created to document information about the specific concepts derived from the core concept. The information includes the code of the specific concept, the code of the parent concept from which the specific concept directly derives, and a label for each language targeted by the ontology (in this case, French and Arabic). Additional attributes such as concept descriptions may also be added if available. For instance, for the core concept "condamnation" (sentence), a dictionary of specific concepts is created to record information about the specific concepts derived from this core concept. I added an Excel sheet for each dictionary of specific concepts to the ontology workbook. For example, for the dictionary of specific concepts for the core concept "condamnation," I created an Excel sheet named "CONDAMNATIONS."

During the phase of extracting specific concepts and organizing them hierarchically, another reading and analysis of the penal code was conducted to study the core concepts in greater detail and extract the specific concepts for each core concept, along with their hierarchical organization. Throughout the analysis of the penal code, terms, their associated concepts, and their hierarchical positions within the ontology framework were identified. Concepts were organized into a hierarchical taxonomy by asking whether being an instance of one class would necessarily make the object an instance of another class. In other words, if class A is a superclass of class B (B is a subclass of A), then every instance of B is also an instance of A.

For example:

- ✓ The statement "Les peines et mesures de sûreté édictées au présent code sont applicables aux majeurs" [The penalties and security measures prescribed in this code apply to adults] identifies two types of "condamnations" [sentences]: "les peines" [penalties] and "les mesures de sûreté" [security measures], which are subclasses (subClassOf) of the class "condamnation" [sentence].
- ✓ The statement "Les peines sont principales ou accessoires" [Penalties are either principal or accessory] identifies two types of "peines" [penalties] : "Les peines principales" [principal penalties] and "Les peines accessoires" [accessory penalties], which are subclasses of the class "peine" [penalty].
- ✓ The statement "Les infractions sont qualifiées crime, délit correctionnel, délit de police ou contravention" [Offenses are categorized as crimes, correctional offenses, police offenses, or infractions] identifies four types of "infractions" [offenses]: "crimes" [crimes], "délits correctionnels" [correctional offenses], "délits de police" [police offenses], and "contraventions" [infractions].

It is important to note that all subclasses of a class inherit the object and data properties of the superclass.

The conceptualization and refinement of the ontology follow a hybrid development process that combines both top-down and bottom-up approaches, explored iteratively as needed. The process begins with a top-down approach, aiming to detail the hierarchy by progressively specializing core concepts into more specific ones. As the hierarchy evolves, it may shift to a bottom-up approach, grouping concepts with common features into more general concepts.

For example, I grouped the concepts "délit correctionnel" [correctional offense] and "délit de police" [police offense] into a more general concept called "délit" [misdemeanor], which is then directly positioned as a subclass of the class "infraction" [offense]. This class is further specialized into three types of "infractions" [offenses]: "crime" [crime], "délit" [misdemeanor], and "contravention" [infraction]. The class "délit" [misdemeanor] itself is further specialized into two subclasses: "délit correctionnel" [correctional offense] and "délit de police" [police offense].

Indeed, the conceptualization phase resembles assembling a puzzle from the pieces provided during knowledge acquisition, which is why much of the knowledge acquisition takes place during conceptualization. Table 4 shows an excerpt from the dictionary of specific concepts for the core concept “condemnation” [sentence].

It is preferable to progressively integrate specific concepts into the ontology skeleton using Protégé, which provides a clear visualization of the hierarchy.

Journal of Theoretical and Applied Information Technology
31st May 2025. Vol.103. No.10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 15 shows the visualization of the hierarchy of specific concepts for the core concept "condamnation" [sentence].

During the study aimed at refining the ontology concepts, relationships between concepts are also refined if new properties emerge. For object properties, the same object properties dictionary is used. For data properties, a separate data properties dictionary is created similarly to provide information about this type of property. The information includes the property code, a label for each language targeted by the ontology (in this case, French and Arabic), the domain, the value type, cardinality, and so on. I added an Excel sheet to the ontology workbook for the data properties dictionary, which I named "DATA_PROPERTIES."

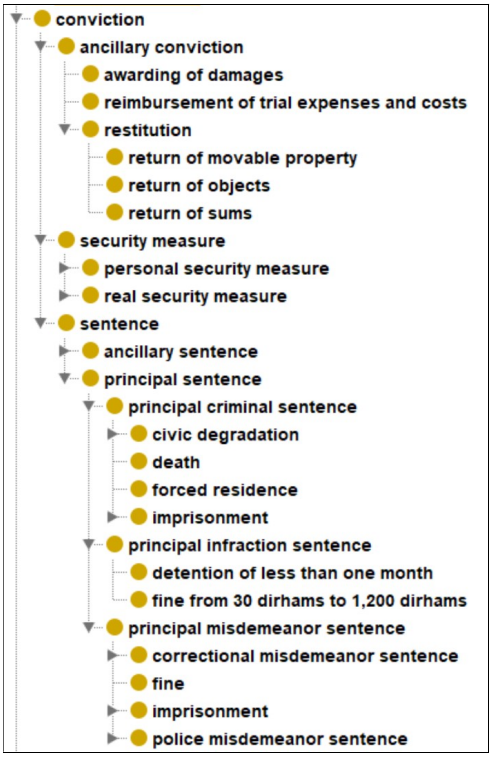


Figure 15: The Visualization Of The Hierarchy Of Specific Concepts For The Core Concept "Condamnation" [Sentence]

Regarding agents, they can be natural persons, legal entities, or groups of individuals. Figure 16 shows the taxonomy of agents, which represent entities within society that must adhere to a code of conduct and may be responsible for offenses.

Table 4: The Dictionary Of Specific Concepts For The Core Concept Sentence

Code	Super-classe	Label Fr	Label Ar	Label En
CONDAMNATION		condamnation	إدانة	conviction
PEINE	CONDAMNATION	Peine	عقوبة	sentence
MESURE_SURETE	CONDAMNATION	mesure de sûreté	تدبير وقائي	security measure
CONDAMNATION_ACCESSOIRE	CONDAMNATION	condamnation accessoire	إدانة إضافية	ancillary conviction
PEINE_PRINCIPALE	PEINE	peine principale	عقوبة أصلية	principal sentence
PEINE_ACCESSOIRE	PEINE	peine accessoire	عقوبة إضافية	ancillary sentence
PEINE_CRIMINELLE	PEINE_PRINCIPALE	peine criminelle principale	عقوبة جنائية أصلية	principal criminal sentence
PEINE_DELICTUELLE	PEINE_PRINCIPALE	peine délictuelle principale	عقوبة جنحية أصلية	principal misdemeanor sentence
PEINE_CONTRAVENTIONNELLE	PEINE_PRINCIPALE	peine contraventionnelle	عقوبة ضبطية أصلية	principal infraction sentence
MORT	PEINE_CRIMINELLE	mort	الإعدام	death
RECLUSION	PEINE_CRIMINELLE	réclusion	السجن	imprisonment
RECLUSION_PERPETUELLE	RECLUSION	réclusion perpétuelle	السجن المؤبد	life imprisonment
RECLUSION_TEMPS	RECLUSION	réclusion à temps	السجن المؤقت	fixed-term imprisonment

Journal of Theoretical and Applied Information Technology
31st May 2025. Vol.103. No.10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 16: The Taxonomy Of Agents

The characteristics of agents that impact convictions are represented by properties. When a property has only a few possible values, it is useful to create an enumerated class (enumeration) to represent these values and explicitly define the class by listing each possible value, which allows the creation of an enumeration. The possible values of an enumerated class are defined as individuals of that class. The following properties are included:

- ✓ "Gender" property: represents the gender of the individual and determines whether the person is male or female. Its value is one of the values from the "gender" enumeration {male, female};
- ✓ "Age" property: represents the age of the individual and determines whether they are of legal age, or a minor under 12 years old or between 12 and 18 years old. Its value is an integer;
- ✓ "Mental state" property: represents the mental state of the individual and determines whether the person is responsible, irresponsible, or partially irresponsible. Its value is one of the values from the "mental state" enumeration {mental impairment, capable of discernment, sane, mental disorder};
- ✓ "Pregnant" property: represents the number of months the woman is pregnant. Its value is an integer;
- ✓ "Postpartum" property: represents the number of days since the woman gave birth. Its value is an integer;
- ✓ "Nationality" property: represents the nationality of the individual and determines whether they are national or not. Its value is one of the values from the "nationality" enumeration {national, binational, foreigner, stateless};

A dictionary of enumerations must be created to fill in the information about the enumerations. I have added an Excel sheet to the ontology workbook for the enumeration dictionary, which I named "ENUMS." Table 5 shows an excerpt from the enumeration dictionary. The enumerated classes are in bold, and the rest are the possible values of the enumerations (individuals).

It is important to note that axioms and rules in ontologies are two complementary concepts used to structure and enrich knowledge bases. Axioms, based on Description Logics (DL), are

Table 5: Excerpt From The Enumeration Dictionary

Code	Type	Label Fr	Label Ar	Label En
NATIONALITE	**ENUM**	**nationalité**	**الجنسية**	**nationality**
NATIONAL	NATIONALITE	national	وطني	national
BINATIONAL	NATIONALITE	binational	ثنائي الجنسية	binational
ETRANGER	NATIONALITE	étranger	أجنبي	foreigner
APATRIDE	NATIONALITE	apatride	عديم الجنسية	stateless
GENRE	**ENUM**	**genre**	**الجنس**	**gender**
HOMME	GENRE	homme	رجل	man
FEMME	GENRE	femme	مرأة	woman
ETAT_MENTAL	**ENUM**	**état mental**	**الحالة العقلية**	**mental state**
ESPRIT_SAIN	ETAT_MENTAL	sain d'esprit	سليم العقل	sane
CAPABLE_DISCERNEMENT	ETAT_MENTAL	capable de discernement	قادر على التمييز	capable of discernment
TROUBLE_MENTAL	ETAT_MENTAL	trouble mental / trouble des facultés mentales	خلل عقلي/ خلل في القوا العقلية	mental disorder / impairment of mental faculties
AFAIBLISSEMENT_MENTAL	ETAT_MENTAL	affaiblissement des facultés mentales	ضعف في القوا العقلية	weakening of mental faculties

Journal of Theoretical and Applied Information Technology
31st May 2025. Vol.103. No.10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

formal logical statements that define relationships, concepts, and constraints in an ontology, while rules, based on Conditional Logics, express conditional relationships or logical actions that are used to infer new knowledge or capture behaviors.

Similarly, two other dictionaries need to be created, one for axioms and another for rules. Therefore, I have added an Excel sheet to the ontology workbook for the axiom dictionary, which I named "AXIOMS," and another sheet for the rule dictionary, which I named "RULES." The enumerations and a set of concepts have been defined with axioms. Table 6 shows an excerpt from the axiom dictionary, where the axioms representing the agents are defined.

Regarding legal entities, which are supposed to represent the law, they can be legal organizations, professional legal persons, and social legal persons. Figure 17 shows the taxonomy of legal entities.

3.5 Identify and create individuals

This step involves identifying and creating instances of the classes, which are called individuals. This can always be done through the analysis of knowledge sources and their textual structures. Instances are the final specification or the most basic form of the classes in the ontology.

Just like for specific concepts, we begin by building dictionaries for individuals, which are somewhat similar to the dictionaries for specific concepts. For each core concept, we create a dictionary of individuals that represent instances of the specific concepts derived from the core concept in question. The individual dictionary allows us to fill in information about the individuals, namely: the individual code, which helps identify the

Table 6: Excerpt from the axiom dictionary

Axiom code	Concepts	Description En	Expression
AXIOM_GENRE	GENRE	The possible values of the "gender" enumerated class are: male or female.	{homme, femme}
AXIOM_ETAT_MENTAL	ETAT_MENTAL	The possible values of the "mental state" enumerated class are: mental impairment, capable of discernment, sane, mental disorder.	{'affaiblissement des facultés mentales', 'capable de discernement', 'sain d\'esprit' , 'trouble mental'}
AXIOM_NATIONALITE	NATIONALITE	The possible values of the "nationality" enumerated class are: national, binational, foreigner, stateless.	{national, binational, étranger, apatride}
AXIOM_FEMME	FEMME	Women are individuals whose gender value is female.	A_GENRE value FEMME
AXIOM_HOMME	HOMME	Men are individuals whose gender value is male.	A_GENRE value HOMME
AXIOM_MINEUR	MINEUR	Minors are individuals whose age is under 18.	AGE some xsd:integer[< 18]
AXIOM_MAJEUR	MAJEUR	Adults are individuals whose age is 18 or older.	AGE some xsd:integer[>=18]
AXIOM_MINEUR_INF_12	MINEUR_INF_12	Minors under 12 are individuals whose age is under 12.	AGE some xsd:integer[<12]
AXIOM_MINEUR_12_18	MINEUR_12_18	Minors between 12 and 18 are individuals whose age is between 12 and 18, exclusive.	(AGE some xsd:integer[>=12]) and (AGE some xsd:integer[>18])
AXIOM_IRRESPONSABLE	IRRESPONSABLE	Irresponsible persons are those with mental disorders and minors under 12 years old.	((A_ETAT_MENTAL value TROUBLE_MENTAL) or (AGE some xsd:integer[<12]))
AXIOM_IRRESPONSABLE_PARTIEL	IRRESPONSABLE_PARTIEL	Partially irresponsible persons are those with mental impairment and minors between 12 and 18 years old.	((AGE some xsd:integer[>=12]) and (AGE some xsd:integer[<18])) or (A_ETAT_MENTAL value AFAIBLISSEMENT_MENTAL)
AXIOM_RESPONSABLE	RESPONSABLE	Responsible persons are those who are sane and capable of discernment, and who are adults.	((A_ETAT_MENTAL value CAPABLE_DISCERNEMENT) or (A_ETAT_MENTAL value ESPRIT_SAIN)) and (AGE some xsd:integer[>=18])

Journal of Theoretical and Applied Information Technology
31st May 2025. Vol.103. No.10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

individual regardless of language; the code of the concept from which the individual is instantiated; a label that defines its linguistic term in each language targeted by the ontology (here French and Arabic); a comment (description) in each language if necessary; and finally, the properties that link the individual to other entities and their values.

Figure 17: The Taxonomy Of Legal Entities

For example, for the core concept "infraction" [offense] we create a dictionary of individuals where we fill in the information for the instances of the specific concepts derived from this core concept "infraction" [offense]. I used an Excel sheet for each individual dictionary in the same ontology workbook. For example, for the dictionary of individuals for the specific concepts of the core concept "infraction" [offense], I created an Excel sheet that I named "INFRACTIONS_INSTANCES."

In the case of the penal code, we are dealing with individuals whose property value depends on the value of another. For example, in the case of the offense "attentat" [assault], we have:

- ✓ "l'attentat est puni de la réclusion perpétuelle s'il est contre le régime" [The assault is punishable by life imprisonment if it is against the regime] which gives the following two triples:
  - ("attentat", "contre", "régime") [("assault", "against", "regime")]
  - ("attentat", "puni de", "réclusion perpétuelle") [("assault", "punished by", "life imprisonment")]
- ✓ "l'attentat est puni de mort s'il est contre la personne du roi avec résultats" [The assault is punishable by death if it is against the king's person with results], which gives the following three triples:
  - ("attentat", "contre la personne de", "roi") [("assault","against the person of", "king")]
  - ("attentat", "résultat", "oui") [("assault", "result", "yes")]
  - ("attentat", "puni de", "mort") [("assault", "punished by", "death")]
- ✓ "l'attentat est puni de la réclusion perpétuelle s'il est contre la personne du roi sans résultats" [The assault is punishable by life imprisonment if it is against the king's person without results], which gives the following three triples:
  - ("attentat", "contre la personne de", "roi") [("assault","against the person of", "king")]
  - ("attentat", "résultat", "non") [("assault", "result", "no")]
  - ("attentat", "puni de", "réclusion perpétuelle") [("assault", "punished by", "life imprisonment")]
- ✓ "l'attentat est puni de la réclusion perpétuelle s'il est contre la personne de l'héritier du trône avec résultats" [The assault is punishable by life imprisonment if it is against the person of the heir to the throne with results], which gives the following three triples:
  - ("attentat", "contre la personne de", "héritier du trône") [("assault", "against the person of", "heir to the throne")]
  - ("attentat", "résultat", "oui") [("assault", "result", "yes")]
  - ("attentat", "puni de", "réclusion perpétuelle") [("assault", "punished by", "life imprisonment")]
- ✓ "l'attentat est puni de la réclusion à temps de 20 à 30 ans s'il est contre la personne de l'héritier du trône sans résultats" [The assault is punishable by a prison sentence of 20 to 30 years if it is against the person of the heir to the throne without results], which gives the following three triples:
  - ("attentat", "contre la personne de", "héritier du trône") [("assault", "against the person of", "heir to the throne")]
  - ("attentat", "résultat", "non") [("assault", "result", "no")]
  - ("attentat", "puni de", "réclusion à temps de 20 à 30 ans") [("assault", "punished by", "prison sentence of 20 to 30 years")]

Thanks to the independence of the concepts in the language ontology, we can represent these different cases of assault with different individuals and unique codes, even if they share the same linguistic term, which is represented by labels for each language. Thus, I created an individual for each case with a unique code, a label for each language, and a comment that helps recognize the individual, as the label is the same, which is "attentat" in French and "الاعتداء" in Arabic. We then define the properties for each individual and therefore for each case. Table 7 shows an excerpt from the dictionary of individuals for the specific concepts of the core concept "infraction" [offense] particularly the individuals that represent the cases of the offense "attentat" [assault]. The labels and comments in Arabic are not included in this table to keep its content concise.

It is important to keep in mind that it is sufficient to assign the value of one property among two inverse properties. The deduction of the second property will be made automatically by the reasoner. For example, if we assign the value "death" to the property "is punished by" for the individual "assault," then we do not need to assign the value "assault" to the property "punishes" for the individual "death," as it will be automatically inferred by the reasoner, since the two properties "is punished by" and "punishes" are defined as inverse properties. Therefore, if we decide to specify the values of the property "is punished by" for the offenses, we do not need to specify the values of the property "punishes" for the convictions.

Table 7: The Dictionary Of Individuals For The Specific Concepts Of The Core Concept "Infraction" [Offense]

Instance code	Type	Label En	Comment En	Property	Value
ATTENTAT_VIE_ROI	ATTENTAT_CRIMINELLE	assault	Assault on the life of the king	CONTRE_VIE	ROI
				PUNI_DE	MORT
ATTENTAT_PERSONNE_ROI	ATTENTAT_CRIMINELLE	assault	Assault on the person of the king	CONTRE_PERSONNE	ROI
				RESULTAT	OUI
				PUNI_DE	MORT
ATTENTAT_PERSONNE_ROI_ECHEC	ATTENTAT_CRIMINELLE	assault	Assault on the person of the king without result	CONTRE_PERSONNE	ROI
				RESULTAT	NON
				PUNI_DE	RECLUSION_PERPETUELLE
ATTENTAT_VIE_HERITIER	ATTENTAT_CRIMINELLE	assault	Assault on the life of the heir to the throne	CONTRE_VIE	HERITIER_TRONE
				PUNI_DE	MORT
ATTENTAT_PERSONNE_HERITIER	ATTENTAT_CRIMINELLE	assault	Assault on the person of the heir to the throne	CONTRE_PERSONNE	HERITIER_TRONE
				RESULTAT	OUI
				PUNI_DE	RECLUSION_PERPETUELLE
ATTENTAT_PERSONNE_HERITIER_ECHEC	ATTENTAT_CRIMINELLE	assault	Assault on the person of the heir to the throne without result	CONTRE_PERSONNE	HERITIER_TRONE
				RESULTAT	NON
				PUNI_DE	RECLUSION_TEMPS_20_30
ATTENTAT_VIE_ROI_FAMILLE	ATTENTAT_CRIMINELLE	assault	Assault on the life of a member of the royal family	CONTRE_VIE	ROI_FAMILLE_MEMBRE
				PUNI_DE	MORT
ATTENTAT_PERSONNE_ROI_FAMILLE	ATTENTAT_CRIMINELLE	assault	Assault on the person of a member of the royal family	CONTRE_PERSONNE	ROI_FAMILLE_MEMBRE
				RESULTAT	OUI
				PUNI_DE	RECLUSION_TEMPS_5_20
ATTENTAT_PERSONNE_ROI_FAMILLE_ECHEC	ATTENTAT_DELICTUELLE	assault	Assault on the person of a member of the royal family without result	CONTRE_PERSONNE	ROI_FAMILLE_MEMBRE
				RESULTAT	NON
				PUNI_DE	EMPRISONNEMENT_2_5
ATTENTAT_REGIME	ATTENTAT_CRIMINELLE	assault	Assault on the regime	CONTRE	REGIME
				PUNI_DE	RECLUSION_PERPETUELLE

We can revisit previous steps and refine the concepts and properties. For example, for the offense "assault," we have cases that are classified as crimes and are punished by criminal penalties, and others classified as misdemeanors and are punished by misdemeanor penalties. As a result, I created two new concepts: “attentat criminal” [criminal assault] as a subclass of the concept “crime” [crime], and “attentat délictuel” [misdemeanor assault] as a subclass of the concept “délit” [misdemeanor]. The individuals "assault" are instances of either "criminal assault" or "misdemeanor assault."

Once the editing of the individual dictionaries is complete, we proceed to integrate them into the ontology with Protégé. Figure 18 shows an excerpt from the individuals of the type “correctional offense against state security.” The display for this excerpt uses the codes (IRIs) of the individuals and not the labels, which may appear similar for some individuals, as explained earlier.

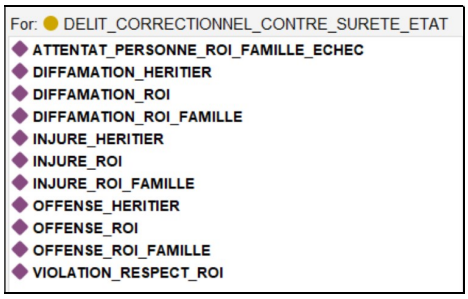


Figure 18: Excerpt Of The Individuals From The Ontology

3.6 Verify the consistency of the ontology and simulate deductive reasoning

The main advantage of using Protégé is the ability to check whether the created ontology contains contradictory definitions, thanks to the inference engine, also called the ontology reasoner. This engine allows for verifying the consistency of the ontology and performing reasoning based on the ontology's knowledge to infer new facts. It can identify various types of ontological relationships, such as transitive, symmetric, inverse, and functional properties, and use them to add new facts.

Therefore, it is important to apply an ontology reasoner on both the ontology and RDF data. This allows for verifying whether all statements and definitions in the ontology are mutually consistent, such as checking that an element is not simultaneously an instance of two classes in a disjoint decomposition. The reasoner also helps deduce additional information. For example, if two properties are inverses and the domain and range of one property are defined, the reasoner knows that the domain of one is the range of the other, and vice versa. This allows the reasoner to infer the domain and range of the inverse property without the user having to define them manually for both properties.

For instance, by defining the property "is punished by" with "offense" as the domain and "conviction" as the range, and the property "punishes" as the inverse of "is punished by," the reasoner infers that the domain of "punishes" is "conviction" and its range is "offense." Also, if two properties are inverses, the user only needs to assert the value of one of the properties, and the inverse value will be automatically inferred by the reasoner. For example, when specifying the triple ("terrorist attack", "is punished by", "death") and knowing that "punishes" is the inverse of "is punished by," the reasoner infers the triple ("death", "punishes", "terrorist attack"). This feature significantly reduces the effort required to populate an ontology, especially with individual data, and that's why running the reasoner frequently can save time and help maintain a valid model. Any information provided by the reasoner instead of the user is highlighted in yellow.

For my part, I used the HermiT reasoner, which must be selected, run, and synchronized through the Reasoner menu in Protégé. To ensure everything is consistent, there should be no errors, otherwise, they need to be corrected.

It’s important to keep in mind that SPARQL ignores information inferred by the reasoner. However, the information inferred by the reasoner can be saved and reloaded so that it is treated the same as user-defined data. This is the solution to use to ensure that the inferred information is not ignored by SPARQL. This solution is described in the article [68].

3.7 Validate and evaluate the ontology

The involvement of experts, whether through their opinions, websites, courses, or videos, helps clarify data that is not well-defined in the knowledge sources. For example, the penal code does not provide information on the Moroccan judicial organization, which was defined and clarified through legal professionals. Protégé also makes it feasible and easy to communicate and exchange with legal professionals and domain experts who are not necessarily ontologists or developers, thanks to its various hierarchical and graphical visualizations that make the ontology

Journal of Theoretical and Applied Information Technology
31st May 2025. Vol.103. No.10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

easily understandable and shareable. This enables domain experts to validate the ontology, as they can confirm the hierarchy of concepts.

Competence questions, in turn, help evaluate the ontology's ability to properly represent the domain knowledge and answer related questions. They represent the ontological commitment made at the beginning of the ontology's development, meaning the ontology must be enriched and corrected until it can answer all competence questions. These questions must be transformed into a formal language, namely SPARQL, to query the ontology with it. Table 8 shows an example of the competence questions defined for the penal code ontology and their equivalent SPARQL queries in the two languages supported by the ontology, which are French and Arabic. Figure 19 shows the results of the query for question Q4 in French and Arabic.

4. CONCLUSION

In this work, I described the development of a multilingual legal ontology in French and Arabic for the Moroccan government's Penal Code

Table 8: Example of the competence questions and their equivalent SPARQL queries

N°	Informal competence questions	Formal competence questions	Valid

Q4	Quelles sont les peines qui répriment un acte de trahison ?
	
	ما هي عقوبات فعل الخيانة؟
		SELECT DISTINCT ?x WHERE {
		?C a ?A.
		?C ?P ?O.
		?O a ?B.
		?C rdfs:label ?x.
		?A rdfs:label "peine"@fr.
		?P rdfs:label "reprime"@fr.
		?B rdfs:label "trahison"@fr.
		FILTER (langMatches(lang(?x), "fr") = true).
		}
		
		SELECT DISTINCT ?x WHERE {
		?C a ?A.
		?C ?P ?O.
		?O a ?B.
		?C rdfs:label ?x.
		?A rdfs:label "عقوبة"@ar.
		?P rdfs:label "يعاقب"@ar.
		?B rdfs:label "الخيانة"@ar.
		FILTER (langMatches(lang(?x), "ar") = true).
		}
			Oui

SPARQL query:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX CP: <http://www.semanticweb.org/kourt/ontologies/2024/10/CodePenal.owl#>
SELECT DISTINCT ?x
WHERE {
   ?C a ?A.
   ?C ?P ?O.
   ?O a ?B.
   ?C rdfs:label ?x.
   ?A rdfs:label "peine"@fr.
   ?P rdfs:label "reprime"@fr.
   ?B rdfs:label "trahison"@fr.
   FILTER (langMatches(lang(?x), "fr") = true).
}
```

"mort"@fr

Figure 19: The query results for the question Q4 in French

(MPCO), by proposing and using a customized methodology, TO-MULTILONTOLOGY, which covers aspects from the specification phase to the validation and evaluation phase, with a detailed implementation process that outlines clearly defined steps to guide and simplify the task of building multilingual ontologies. This methodology also addresses one of the main obstacles to effective knowledge sharing: the inadequate documentation of existing ontologies, by providing powerful tools and models that both document the ontology and guide its development.

The TO-MULTILONTOLOGY methodology proposed for the development of multilingual ontologies consists of seven steps: 1) Establish the ontology charter – Specification; 2) Build the ontology skeleton; 3) Define the basic properties between the core concepts of the ontology skeleton; 4) Conceptualize and refine the ontology; 5) Identify and create the individuals; 6) Verify the consistency of the ontology and simulate deductive reasoning; and 7) Validate and evaluate the ontology.

The multilingual legal ontology MPCO (Multilingual Penal Code Ontology) is built in French and Arabic and covers the Moroccan government's Penal Code. It can play an important role in information retrieval through question-answering systems and in learning the knowledge of the Penal Code. It can also serve as a reference for the development of similar criminal law ontologies.

REFERENCES:

[1] M. Debellis, "A practical guide to building OWL ontologies using Protégé 5.5 and plugins", Dostupno: https://www. michaeldebellis. com/post/new-protege-pizzatutorial/.[Pristupljeno: 12. rujna 2024.], 2021.

[2] "Ontologie (philosophie)". Available: https://fr.wikipedia.org/wiki/Ontologie_(philosophie).

[3] T. R. Gruber, "A translation approach to portable ontology specifications", Knowledge acquisition, 1993, vol. 5, n° %12, p. 199–220.

[4] T. R. Gruber, "Toward principles for the design of ontologies used for knowledge sharing?", International journal of human-computer studies, 1995, vol. 43, n° %15-6, p. 907–928.

[5] W. N. Borst, "Construction of engineering ontologies for knowledge sharing and reuse", 1997.

[6] R. Studer, V. R. Benjamins et D. Fensel, "Knowledge engineering: Principles and methods", Data & knowledge engineering, 1998, vol. 25, n° %11-2, p. 161–197.

[7] M. Uschold et M. Gruninger, "Ontologies: Principles, methods and applications", The knowledge engineering review, 1996, vol. 11, n° %12, p. 93–136.

[8] N. F. Noy et D. L. McGuinness, "Ontology development 101: A guide to creating your first ontology", Stanford knowledge systems laboratory technical report, pp. KSL-01-05, 2001.

[9] A. Gangemi, N. Guarino, C. Masolo, A. Oltramari et L. Schneider, "Sweetening Ontologies with DOLCE", Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web, vol. 2473, A. Gómez-Pérez et V. R. Benjamins, Éds., Berlin, Heidelberg, Springer Berlin Heidelberg, 2002, pp. 166-181.

[10] A. Gangemi, N. Guarino, C. Masolo et A. Oltramari, "Sweetening wordnet with dolce", AI magazine, 2003, vol. 24, n° %13, p. 13–13.

[11] J. Breuker et R. Hoekstra, "Core concepts of law: taking common-sense seriously", Proceedings of formal ontologies in information systems (FOIS-2004), 2004.

[12] J. Breuker et R. Hoekstra, "Epistemology and ontology in core ontologies: FOLaw and LRI-Core, two", Proceedings of EKAW Workshop on Core ontologies [Internet]. Northamptonshire, UK: Sun SITE Central Europe, 2004.

[13] A. Gangemi, M.-T. Sagri et D. Tiscornia, "A Constructive Framework for Legal Ontologies", Law and the Semantic Web, vol. 3369, V. R. Benjamins, P. Casanovas, J. Breuker et A. Gangemi, Éds., Berlin, Heidelberg, Springer Berlin Heidelberg, 2005, pp. 97-124.

[14] R. Hoekstra, J. Breuker, M. Di Bello et A. Boer, "The lkif core ontology of basic legal concepts", LOAIT, 2007, vol. 321, p. 43–63.

[15] R. Hoekstra, J. Breuker, M. Di Bello et A. Boer, "LKIF core: Principled ontology development for the legal domain", Law, ontologies and the semantic web, IOS Press, 2009, p. 21–52.

[16] S. Després et S. Szulman, "Choisir une ontologie noyau en vue de sa réutilisation Une stratégie d'évaluation", Journées Francophone sur les Ontologies, 2009.

[17] N. Casellas, "Legal ontology engineering: Methodologies, modelling trends, and the ontology of professional judicial knowledge", Springer Science & Business Media, 2011, vol. 3.

[18] C. M. de Oliveira Rodrigues, F. L. G. de Freitas, E. F. S. Barreiros, R. R. de Azevedo et A. T. de Almeida Filho, "Legal ontologies over time: A systematic mapping study", Expert Systems with Applications, 2019, vol. 130, p. 12–30.

[19] V. Leone, L. Di Caro and S. Villata, "Taking stock of legal ontologies: a feature-based comparative analysis", Artificial Intelligence and Law, 2020, vol. 28, no. 2, pp. 207-235.

[20] S. Steyskal et A. Polleres, "Defining expressive access policies for linked data using the ODRL ontology 2.0", Proceedings of the 10th International Conference on Semantic Systems, Leipzig Germany, 2014.

[21] "LDR". Available: http://oeg-dev.dia.fi.upm.es/licensius/static/ldr/.

[22] "ccREL". Available: https://www.w3.org/Submission/ccREL/.

[23] "L4LOD". Available: http://ns.inria.fr/l4lod/v2/l4lod_v2.html

[24] I. Distinto, M. d'Aquin et E. Motta, "LOTED2: An ontology of European public procurement notices", Semantic Web, 2016, vol. 7, n° %13, p. 267–293.

[25] J. F. Muñoz-Soro, G. Esteban, O. Corcho et F. Serón, "PPROC, an ontology for transparency in public procurement", Semantic Web, 2016, vol. 7, n° %13, p. 295–309.

[26] C. Bartolini, R. Muthuri et C. Santos, "Using Ontologies to Model Data Protection Requirements in Workflows", New Frontiers in Artificial Intelligence, vol. 10091, M. Otake, S. Kurahashi, Y. Ota, K. Satoh et D. Bekki, Éds., Cham, Springer International Publishing, 2017, pp. 233-248.

[27] H. J. Pandit, K. Fatema, D. O'Sullivan et D. Lewis, "GDPRtEXT - GDPR as a Linked Data Resource", The Semantic Web, vol. 10843, A. Gangemi, R. Navigli, M. Vidal, P. Hitzler, R. Troncy, L. Hollink, A. Tordai et M. Alam, Éds., Cham, Springer International Publishing, 2018, pp. 481-495.

[28] A. Oltramari, D. Piraviperumal, F. Schaub, S. Wilson, S. Cherivirala, T. B. Norton, N. C. Russell, P. Story, J. Reidenberg et N. Sadeh, "PrivOnto: A semantic framework for the analysis of privacy policies", Semantic Web, 2018, vol. 9, n° %12, p. 185–203.

[29] H. Haapio, M. Hagan, M. Palmirani et A. Rossi, "Legal design patterns for privacy", Data protection/LegalTech. Proceedings of the 21th international legal informatics symposium IRIS, 2018.

[30] "Eurovoc". Available: https://publications.europa.eu/en/web/eu-vocabularies/th-dataset/-/resource/dataset/eurovoc

[31] M. Palmirani, G. Governatori, A. Rotolo, S. Tabet, H. Boley et A. Paschke, "LegalRuleML: XML-Based Rules and Norms", Rule-Based Modeling and Computing on the Semantic Web, Berlin, Heidelberg, Springer Berlin Heidelberg, 2011, vol. 7018, pp. 298-312.

[32] T. Athan, G. Governatori, M. Palmirani, A. Paschke et A. Wyner, "LegalRuleML: Design Principles and Foundations", Reasoning Web. Web Logic Rules, Cham, Springer International Publishing, 2015, vol. 9203, pp. 151-188.

[33] E. T. Force, "ELI implementation methodology: good practices and guidelines", Publications Office, 2018.

[34] F. Gandon, G. Governatori et S. Villata, "Normative requirements as linked data", Legal Knowledge and Information Systems, IOS Press, 2017, p. 1–10.

[35] K. Dhouib et F. Gargouri, "Legal application ontology in Arabic", Fourth International Conference on Information and Communication Technology and Accessibility (ICTA), 2013.

[36] J. Breuker, A. Valente et R. Winkels, "Use and Reuse of Legal Ontologies in Knowledge Engineering and Information Management", Law and the Semantic Web, Berlin, Heidelberg, Springer Berlin Heidelberg, 2005, vol. 3369, pp. 36-64.

[37] O. Corcho, M. Fernández-López, A. Gómez-Pérez et A. López-Cima, "Building Legal Ontologies with METHONTOLOGY and WebODE", Law and the Semantic Web, Berlin, Heidelberg, Springer Berlin Heidelberg, 2005, vol. 3369, pp. 142-157.

[38] M. Gruninger, "Methodology for the design and evaluation of ontologies", Proc. IJCAI'95, Workshop on Basic Ontological Issues in Knowledge Sharing, 1995.

[39] M. Uschold et M. King, "Towards a methodology for building ontologies", Citeseer, 1995.

[40] A. Gómez-Pérez, M. Fernández et A. d. Vicente, "Towards a method to conceptualize domain ontologies", 1996.

[41] M. Fernández-López, A. Gómez-Pérez et N. Juristo, "Methontology: from ontological art towards ontological engineering", 1997.

[42] S. Staab, R. Studer, H.-P. Schnurr et Y. Sure, "Knowledge processes and ontologies", IEEE Intelligent systems, 2001, vol. 16, n° %11, p. 26–34.

[43] M. F. López, A. Gómez-Pérez, J. P. Sierra et A. P. Sierra, "Building a chemical ontology using methontology and the ontology design environment", IEEE Intelligent Systems and their applications, 1999, vol. 14, n° %11, p. 37–46.

[44] S. Boyce et C. Pahl, "Developing domain ontologies for course content", Journal of Educational Technology & Society, 2007, vol. 10, n° %13, p. 275–288.

[45] Y. Alfaifi, "Ontology development methodology: A systematic review and case study", 2022 2nd International Conference on Computing and Information Technology (ICCIT), 2022.

[46] T. Mondary, S. Després, A. Nazarenko et S. Szulman, "Construction d'ontologies à partir de textes: la phase de conceptualisation", 19èmes Journées Francophones d'Ingénierie des Connaissances (IC 2008), 2008.

[47] P. Cimiano et J. Völker, "Text2onto: A framework for ontology learning and data-driven change discovery", Natural Language Processing and Information Systems, Berlin, Heidelberg, Springer Berlin Heidelberg, 2005, vol. 3513, pp. 227-238.

[48] B. Fortuna, M. Grobelnik et D. Mladenic, "Semi-automatic data-driven ontology construction system", Proceedings of the 9th International multi-conference Information Society IS-2006, Ljubljana, Slovenia, 2006.

[49] N. Aussenac-Gilles, S. Despres et S. Szulman, "The TERMINAE Method and Platform for Ontology Engineering from Texts", 2008.

[50] M. Silberztein, "Formalizing natural languages: The NooJ approach", John Wiley & Sons, 2016.

[51] G. Lame, "Using NLP Techniques to Identify Legal Ontology Components: Concepts and Relations", Law and the Semantic Web: Legal Ontologies, Methodologies, Legal Information Retrieval, and Applications, Berlin, Heidelberg, Springer, 2005, pp. 169-184.

[52] T. Berners-Lee, J. Hendler et O. Lassila, "Web Semantic", Scientific American, 2001, vol. 284, n° %15, p. 34–43.

[53] A. J. Gerber, A. Barnard et A. J. Van der Merwe, "Semantic Web status model", 2006.

[54] H. Jabeen, D. Graux et G. Sejdiu, "Chapter 7 Scalable Knowledge Graph Processing Using SANSA", Knowledge Graphs and Big Data Processing, Cham, Springer International Publishing, 2020, vol. 12072, pp. 105-121.

[55] T. Berners-Lee, R. Fielding et L. Masinter, "Uniform resource identifier (URI): Generic syntax", 2005.

[56] "W3C – XML". Available: https://www.w3.org/TR/xml/

[57] "W3C – RDF". Available: https://www.w3.org/TR/rdf12-concepts/

[58] O. Lassila, "Resource description framework (RDF) model and syntax specification, W3C", 1999.

[59] B. Mcbride, "The Resource Description Framework (RDF) and its Vocabulary Description Language RDFS", Berlin, Heidelberg, 2004.

[60] D. Brickley et R. V. Guha, "RDF Shema", 1999. Available: https://www.w3.org/TR/rdf-schema/.

[61] G. Antoniou et F. V. Harmelen, "Web Ontology Language: OWL", Handbook on Ontologies, 2003.

[62] D. L. McGuinness et F. V. Harmelon, "OWL Web Ontology Language Overview", 2004.

[63] A. Polleres, «SPARQL rules,» 2006.

[64] E. Prud'hommeaux et A. Seaborne, "SPARQL Query Language for RDF", 2008. Available: https://www.w3.org/TR/rdf-sparql-query/.

[65] B. DuCharme, Learning SPARQL, O'Reilly Media, Inc., 2013.

[66] V. Jain et M. Singh, "Ontology development and query retrieval using protégé tool", International Journal of Intelligent Systems and Applications, 2013, vol. 9, n° %19, p. 67–75.

[67] C. Bezerra, F. Freitas et F. Santana, "Evaluating ontologies with competency questions", 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2013, vol. 3, p. 284–285.

[68] M. Debellis, "Protege Best Practices: Export Inferred Axioms", Available: https://www.michaeldebellis.com/post/export-inferred-axioms.