

Autores 3.0: Online Intellectual Property Registry using Semantic Technology



Basic Ontology

0.12

Contents

Requirements Specification.....	2
Reused Resources.....	7
Conceptual Model	15
Implementation	18
References	28

Requirements Specification

The Basic ontology has been developed to provide basic answers regarding the intellectual property of a work.

The first step to develop the ontology is to explicitly document the requirements it should satisfy. Ontology requirements specifications will be made using a tabular format [5] showing:

- Purpose: The main general goal of the ontology. In other words, the main function or role that the ontology should have.
- Scope: The general coverage and the degree of detail that the ontology should have.
- Implementation Language: The formal language that the ontology should have.
- Intended End-Users: The intended end-users expected for the ontology.
- Intended Uses: The intended uses expected for the ontology.
- Ontology Requirements:
 - Non-Functional Requirements: The general requirements or aspects that the ontology should fulfill, including optionally priorities for each requirement.
 - Functional requirements: The content specific requirements that the ontology should fulfill, in the form of groups of competency questions and their answers, including optionally priorities for each group and for each competency question.
- Pre-Glossary of Terms:
 - Terms from Competency Questions: The list of terms included in the competency questions and their frequencies.
 - Terms from Answers: The list of terms included in the answers and their frequencies.
 - Objects: The list of objects included in the competency questions and in their answers.

Table 1 formally specifies the requirements of the Basic ontology.

Autores 3.0	
Basic ontology	
1	Purpose
	<i>This ontology is developed to provide a vocabulary about intellectual property domain agreed by general consent that could be used by users and computer programs to model works and its exploitation rights transmitted by means of licenses and other cession agreements, so it can be known which basic actions can be legally carried out with a specific work.</i>
2	Scope
	<i>The ontology will only be focused on the intellectual property domain limited to licenses, rights and special conditions. Specific national legislations will not be taken into account due to its generalist vocation.</i> <i>The granularity level is directly related to the competency questions and the identified concepts.</i>
3	Implementation Language
	<i>The ontology will be implemented in OWL DL due to its enormous power of expressivity.</i>
4	Intended End-Users
	<i>User 1. P2P programs to know if certain actions (download, reproduction, interactively available...) regarding a specific work are legal.</i> <i>User 2. Person who needs to know the legal uses that can carry out with a specific work.</i> <i>User 3. Internet user or webmaster who wants to know the uses of a work he has just downloaded.</i> <i>User 4. Computer program who needs to know the intellectual property aspects related to a work.</i>
5	Intended Uses
	<i>Use 1. P2P program checking if it can allow the download of a work.</i> <i>Use 2. P2P program checking if it can allow the reproduction of a work or making it available.</i> <i>Use 3. Person checking the legal uses of a specific work.</i> <i>Use 4. Automatic discovery of plagiarism or other illegal uses of a work in Internet.</i> <i>Use 5. Automatic warning to users about the illegal use of a specific work.</i>

Autores 3.0	
Basic ontology	
6	Ontology Requirements
	a. Non-Functional Requirements
	<p><i>NFR 1. The ontology will be documented in English and annotated in English and Spanish.</i></p> <p><i>NFR 2. The ontology will be documented in English and Spanish.</i></p> <p><i>NFR 3. The terminology will be checked on WordNet.</i></p>
	b. Functional Requirements: Groups of Competency Questions
	<p style="text-align: center;">GCQ 1. Works (1 CQ)</p> <p>CQ 1. ¿Which are the titles of the works created by Thomas Schröder? Akarysatos, Meine kleine Tochter im Zoo and La mirada difusa.</p>
	<p style="text-align: center;">GCQ 2. Rights and Special conditions (2 CQ)</p> <p>CQ 2. ¿Can I create derivative works from “Akarys91”? Yes</p> <p>CQ 3. ¿Can I use the work “Akarisene” with commercial purposes? Yes</p>
	<p style="text-align: center;">GCQ 3. People (1 CQ)</p> <p>CQ 4. ¿What is the author’s name of the sculpture “Akarys91”? Thomas Schröder</p>
	<p style="text-align: center;">GCQ 4. Registry (1 CQ)</p> <p>CQ 5. ¿What is the URL where can be checked the registry of the work “Akarys91”? http://www.safecreative.org/work/1003015656515</p>

Autores 3.0	
Basic ontology	
7	Pre-Glossary of Terms
a. Terms from Competency Questions (CQ)	
<i>Work (50%)</i> <i>Right (12.5%)</i> <i>Special Condition (12.5%)</i>	<i>Registry (12.5%)</i> <i>Author (12.5%)</i>
b. Terms from Answers	
<i>Work (75%)</i>	<i>Author (25%)</i>
c. Objects	
Rights	
Display Right Reproduction Right Public Performance Right Transformation Right Communication Right Distribution Right	
Licenses	
Apache BSD GFDL (<i>GNU Free Documentation License</i>) GPL (<i>General Public License</i>) Licencias Creative Commons: CC-BY, CC-BY-NC, CC-BY-NC-ND, C-BY-NC-SA, CC-BY-ND y CC-BY-SA. L-GPL (<i>Lesser General Public License</i>) MIT Mozilla Public License Licencias personalizadas	

Autores 3.0	
Basic ontology	
7	Pre-Glossary of Terms
	c. Objects
	Special Conditions
	Commercial use allowed.
	Commercial use not allowed.
	Derivative works must be licensed with the same license of the original work (<i>Share-alike</i>).
	People
	Author

Table 1 – Basic Ontology Requirements Specification

Reused Resources

Three ontological resources have been reused during the development of the Basic ontology. Classes, object properties and data properties of each reused ontological resource will be shown, showing specifically using a red rectangle which classes and properties have been reused.

First of all, it has been reused the *FOAF* [1] (acronym of *Friend of a friend*) ontology. This ontology was created to describe people, its activities and its relationships with things or other people. Illustration 1 shows the classes defined by this ontology.

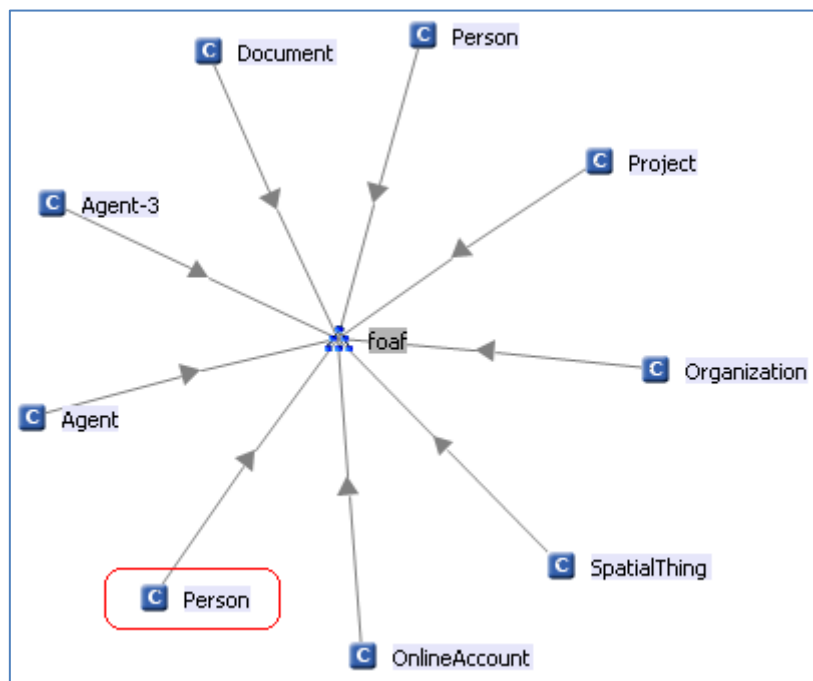


Illustration 1 – Classes defined in the FOAF ontology

Class *Person* has been reused by the Basic ontology to be able to represent authors as persons (that have created some works). Illustration 2 shows the object properties defined by the *FOAF* ontology.

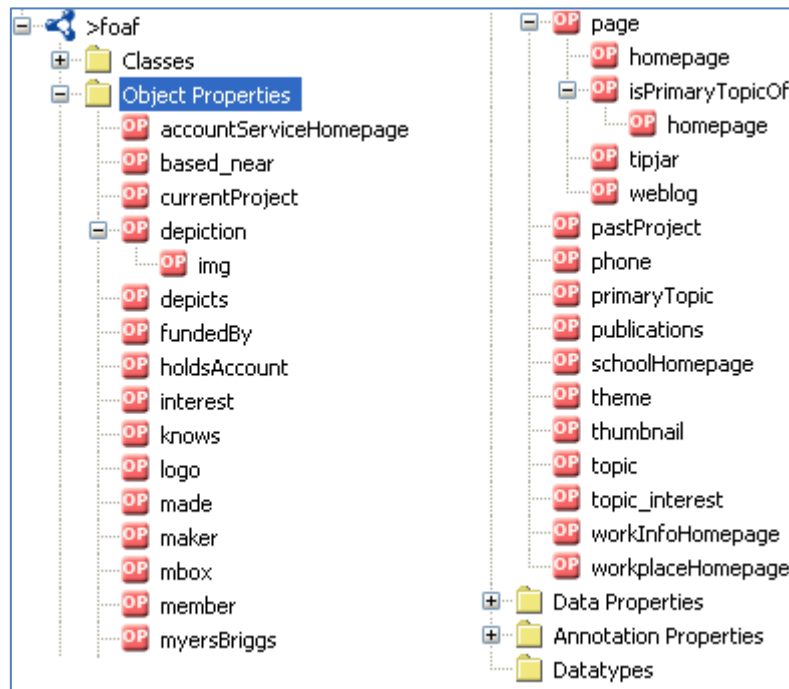


Illustration 2 – Object properties defined in the FOAF ontology

The Basic ontology has not reused any object property. These properties are useful to represent that a creator knows another one, etc. but they are not useful to represent the more basic aspects about the intellectual property of a work.

Lastly, Illustration 3 shows the data properties defined by the *FOAF* ontology. This type of properties is useful to describe people (in this case, to describe creators).

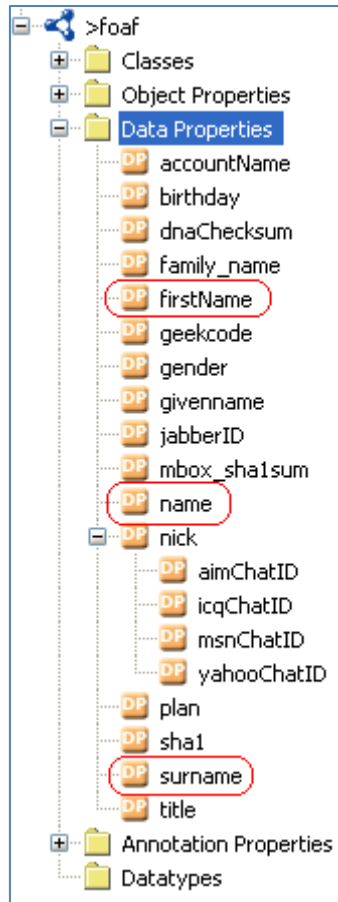


Illustration 3 – Data properties defined in the FOAF ontology

The properties *name*, *firstName* and *surname* have been reused to represent the full name, first name and surname of the creator, respectively.

Secondly, it has also been reused the Simple Dublin Core [2] ontology. This ontology was created in order to describe any type of information resource and it allows describing online resources to make them easily traceable. Illustration 4 shows the classes defined by this ontology.

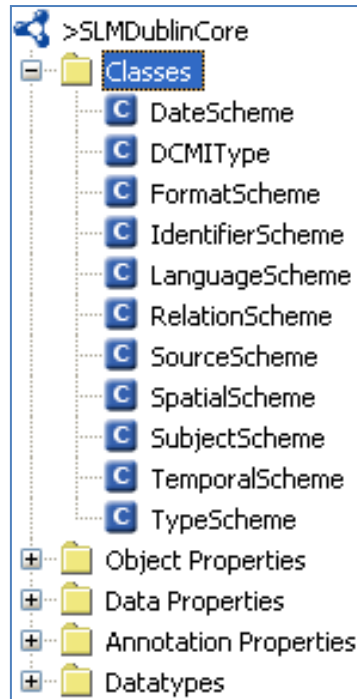


Ilustración 4 – Classes defined in the Simple DC ontology

These classes are very generic and therefore they are not useful for the ontology to be built. The object properties defined by this ontology are shown in Illustration 5.

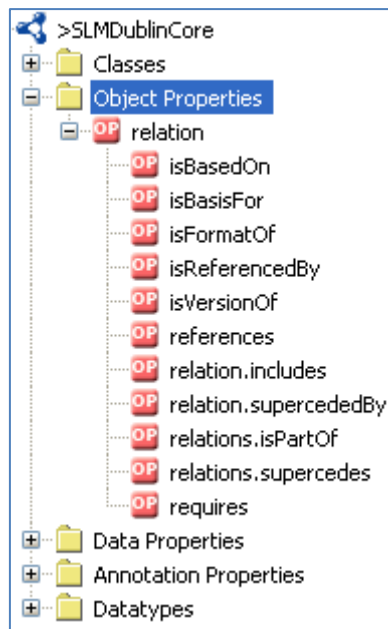


Illustration 5 – Object properties defined in the Simple DC ontology

As can be seen, none of these object properties have been reused due to the fact that it is beyond the goals of this project to model the relationships between works. The data properties defined by this ontology are shown in Illustration 6.

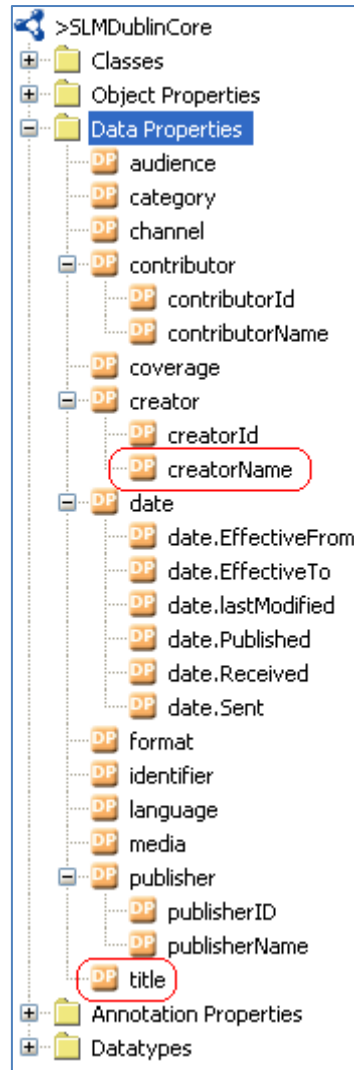


Illustration 6 – Data properties defined in the Simple DC ontology

The properties *creatorname* and *title* have been reused to represent the creator’s full name and the title of the work. It could be said that the property *creatorname* of the Simple DC ontology is equivalent to the property *name* of the FOAF ontology.

Thirdly, the ontology *Qualified Dublin Core* has also been reused. The *Qualified DC* ontology is an extension of the *Simple DC* ontology, and it has been obtained in [3]. The main difference between them lies in the definition of the classes *Audience*, *Provenance* and *RightsHolder*, that allow representing whom a work has been published for or could be useful to, where the work comes from and who holds their rights.

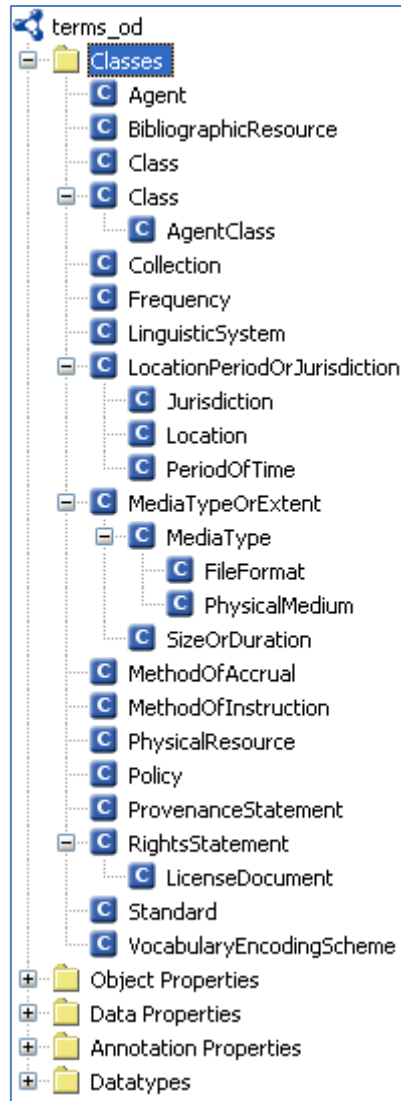


Illustration 7 – Classes defined in the Qualified DC ontology

The Basic ontology, contrary to the Registry ontology, has not reused class *Jurisdiction*. Illustration 8 shows the object properties of this ontology.

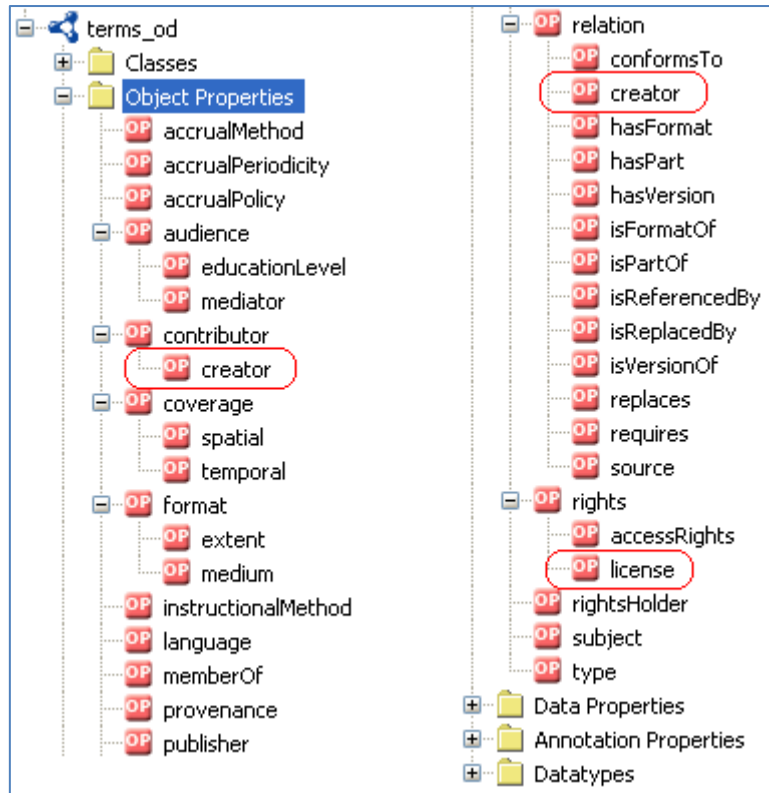


Illustration 8 – Object properties defined in the Qualified DC ontology

The properties *creator* are equivalent and are used to relate each work with its creator, while property *license* relate *Work* and *License* objects and will indicate which license a work is registered under.

Illustration 9 shows the data properties defined by this ontology.

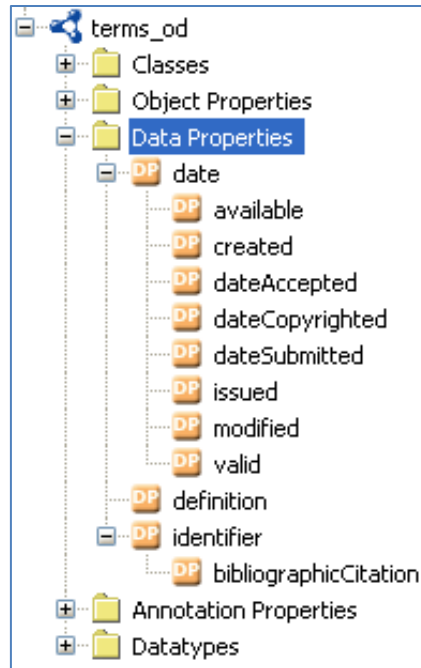


Illustration 9 – Data properties defined in the Qualified DC ontology

As can be seen, no data property has been reused by the Basic ontology.

Conceptual Model

Illustration 10 shows the importation hierarchy of the ontology network generated by *Neon Toolkit*. It can be seen that *basic* is the main ontology.

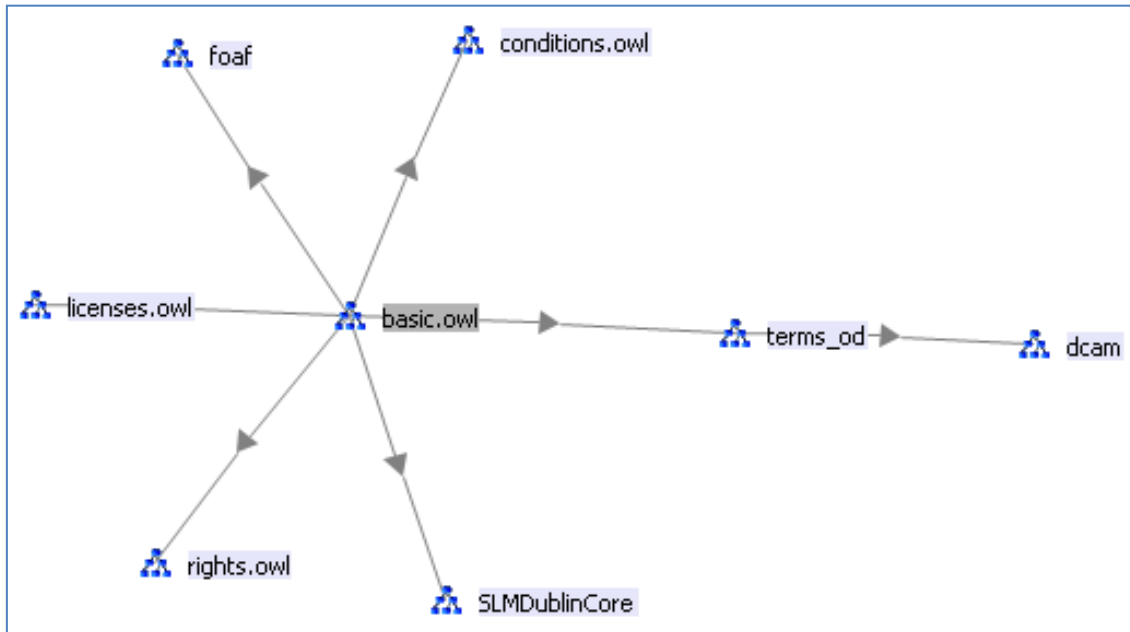


Illustration 10 – Importation hierarchy of the ontology network

The Basic ontology models authors, works, licenses, rights and special conditions. However, this ontology is only populated with individuals from the class *Person* and — to obtain a more modular result — the ontologies *licenses*, *rights* and *conditions* are populated with the remaining individuals and are imported by the *basic* ontology.

Basic is the main ontology and imports the remaining ones, including *FOAF* [1], *Simple Dublin Core* [2] and *Qualified Dublin Core* [3] — some of their resources have been reused.

The ontology has been developed in English, documented in English and Spanish and fulfills the five specified requirements.

Illustration 11 shows the class diagram¹ of the Basic ontology. A better understanding of the ontology can be achieved with a class diagram because it explicitly shows how its classes relate each other.

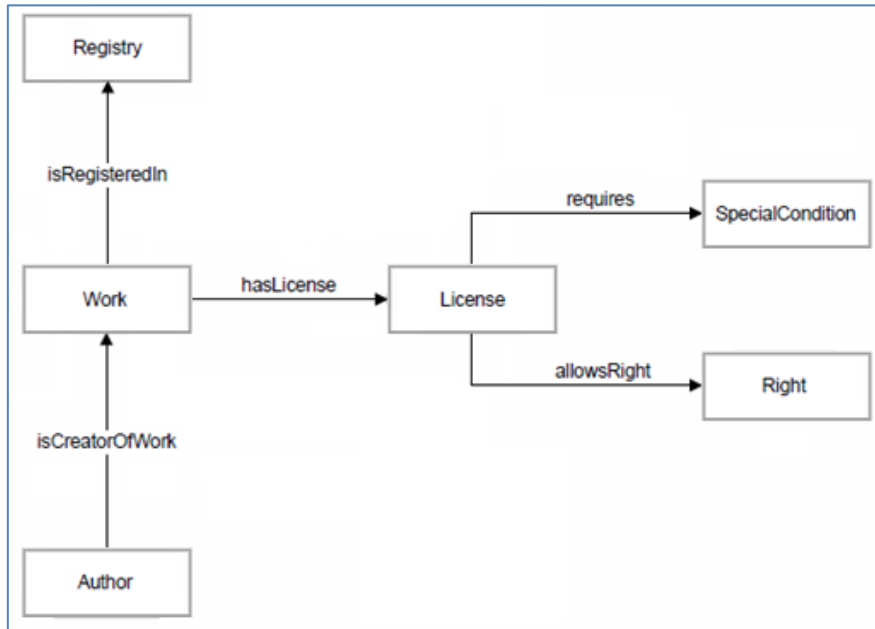


Illustration 11 – Class diagram of the Basic ontology

The main classes of the ontology are *Work* and *License*. A work is registered in a registry, has been created by an author and is registered under a specific license that allows some rights and requires certain special conditions.

The object properties shown in the class diagram — together with its inverse properties — are the more relevant defined in this ontology. On the other hand, the data properties to define author’s personal data, name and URL of the registry, title of the work and URL where the full legal text of the license can be found are the most relevant ones.

The most relevant individuals defined in this ontology are those to define each type of license, special condition and right taken into account in this ontology.

Illustrations 12 to 14 show the hierarchies created for classes *License*, *Right* and *Special Condition*, respectively.

¹ Rectangles represent classes and directed arrows between classes indicate that they are related through an object property whose names are on the arrows. The origin of the arrow is the domain of the property and is directed towards the class that is the range of the property.

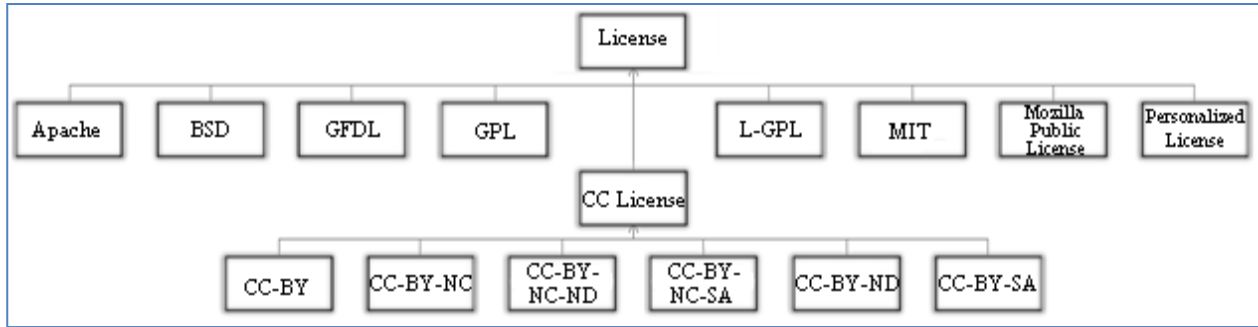


Illustration 12 – Taxonomy of the class License

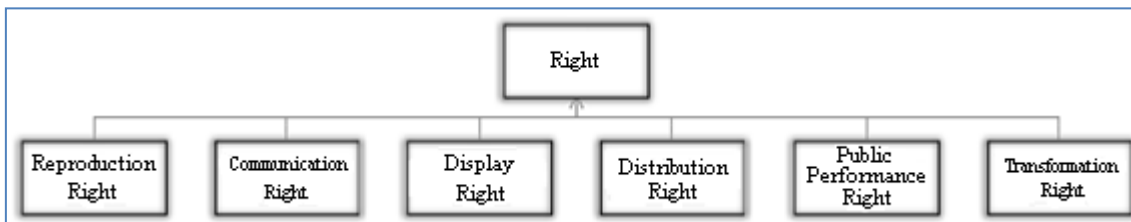


Illustration 13 – Taxonomy of the class Right

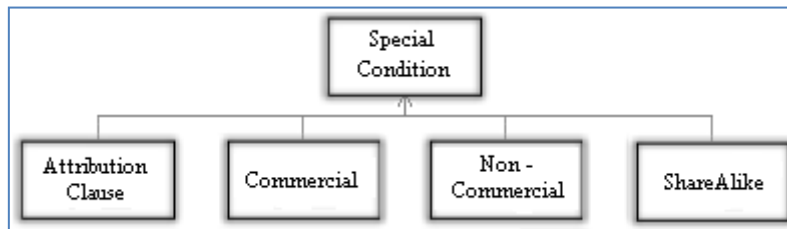


Illustration 14 – Taxonomy of the class Special Condition

Implementation

The ontology has been developed using *NeOn Toolkit* [4] v3.2.1 and *gOntt*, *Relationship Visualization*, *RaDON* and *XD Tools* plug-ins and some documentation² has also automatically generated using *OWLDoc* plug-in.

Several screenshots of the Basic ontology from the *NeOn Toolkit* will be shown in this section. Besides, OWL code will also be shown to provide a more detailed view about this ontology.

The five ontologies signaled in Illustration 15 have been reused. *Core*, *SLMDublinCore* and *foaf* ontologies have been reused directly, while *dcam* and *terms_od* have been reused indirectly due to the fact that these ontologies are imported by the former ones.

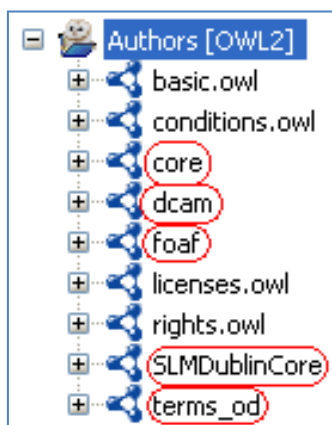


Ilustración 15 – Reused ontologies in the Basic ontology

Illustration 16 shows the classes defined by the Basic ontology (those that have been reused are signaled).

² The documentación is available in [6].

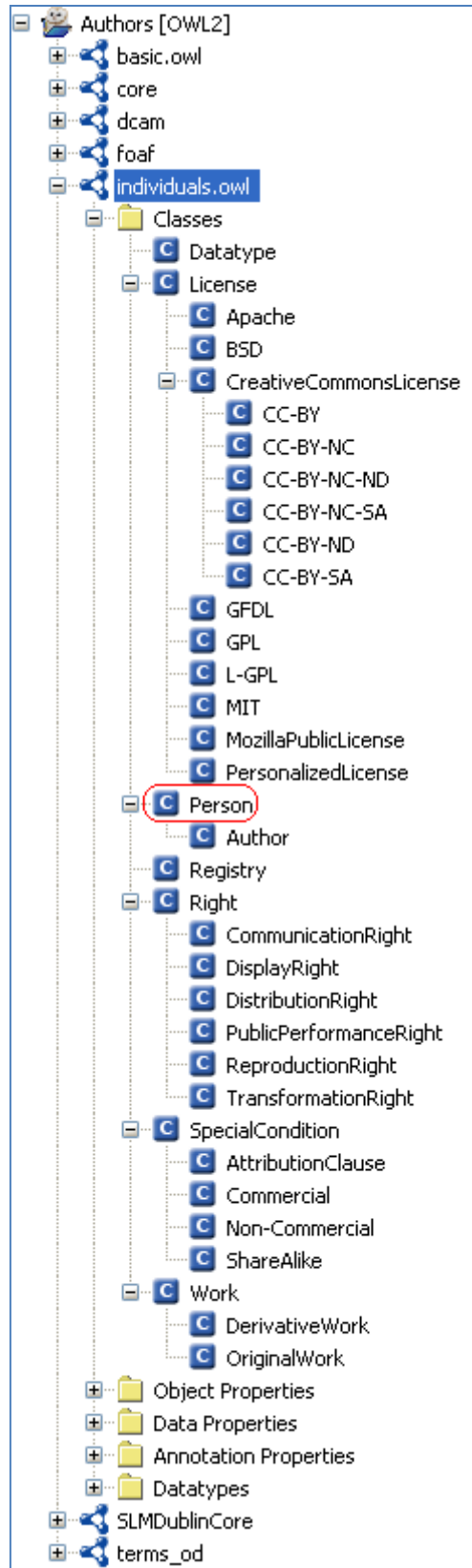


Illustration 16 – Classes defined in the Basic ontology

Illustration 17 shows the object properties defined by the Basic ontology (those that have been reused are signaled).

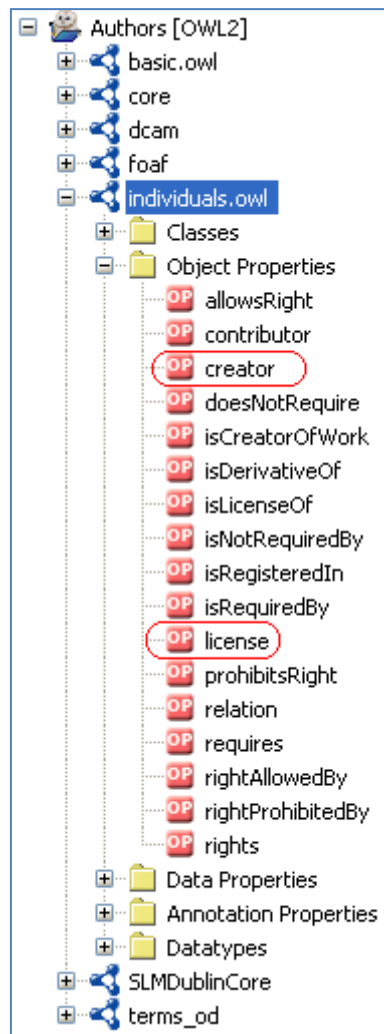


Illustration 17 – Object properties defined in the Basic ontology

Illustration 18 shows the data properties defined by the Basic ontology (those that have been reused are signaled).

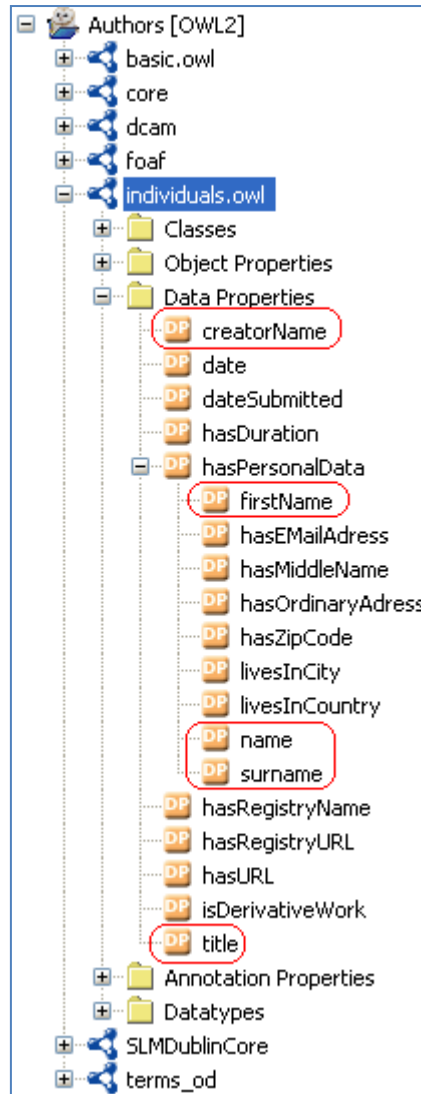


Illustration 18 – Data properties defined in the Basic ontology

Illustration 19 shows the relationships of the class *Work*, visualized using *Relationship visualization* plug-in of *Neon Toolkit*.

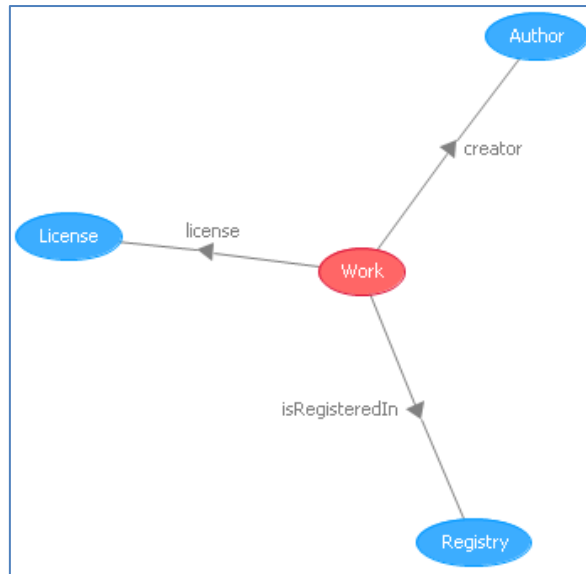


Illustration 19 – Relations of the class *Work* in the Basic ontology

It can be observed that a work is registered under a certain license, is registered in a specific registry and was created by a specific author.

Relationships of the class *License* are shown in Illustration 20.

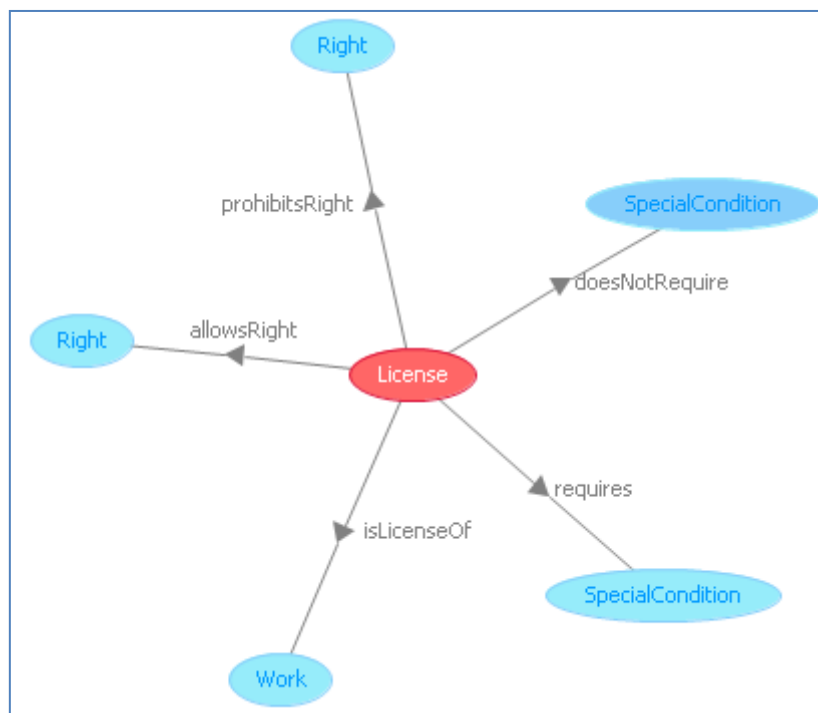


Illustration 20 – Relations of the class *License* in the Basic ontology

It can be observed that a work is licensed under a specific license and each license allows certain rights and requires some special conditions. Lastly, due to the close world assumption, it is explicitly indicated, which special conditions are not required and which rights are not allowed.

Illustration 21 shows the Manchester Syntax of the class *License*.

▼ Frame Representation in Manchester Syntax

Class: basic:License

Annotations:
 rdfs:comment "A legal document giving official permission to do something."@en,
 rdfs:comment "Documento en que consta la autorización de una determinada actividad o permisos."@es,
 rdfs:label "licencia"@es

EquivalentTo:
 basic:allowsRight **some** basic:Right,
 basic:doesNotRequire **some** basic:SpecialCondition,
 basic:prohibitsRight **some** basic:Right,
 basic:requires **some** basic:SpecialCondition

SubClassOf:
 owl:Thing

Illustration 21 – Class License in the Basic ontology using the Manchester syntax

It can also be observed that this class is documented with a definition both in English and Spanish and has class expressions to model that a license allows some rights and requires certain special conditions.

Illustration 22 shows OWL code of the class *License*.

```

<owl:Class rdf:about="#License">
  <rdfs:label xml:lang="es">licenciatarario</rdfs:label>
  <owl:equivalentClass>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#allowsRight"/>
      <owl:someValuesFrom rdf:resource="#Right"/>
    </owl:Restriction>
  </owl:equivalentClass>
  <owl:equivalentClass>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#hasSpecialCondition"/>
      <owl:someValuesFrom rdf:resource="#SpecialCondition"/>
    </owl:Restriction>
  </owl:equivalentClass>
  <rdfs:subClassOf rdf:resource="owl:Thing"/>
  <rdfs:comment xml:lang="en">
    >A legal document giving official permission to do something.</rdfs:comment>
  <rdfs:comment xml:lang="es">
    >Documento en que consta la autorizaci&#243;n de una determinada actividad o permisos.</rdfs:comment>
</owl:Class>
    
```

Illustration 22 – Simplified OWL code of the class License in the Basic ontology

The basic Ontology contains only individuals of the class *Person*, that is, *Author_1*. In order to obtain a more modular result, the remaining individuals have been defined in the ontologies *licenses*, *rights* and *specialConditions*. Besides, these individuals have been linked, using the label *sameAs*, with a set of individuals from *DBpedia* [7] and *Freebase* [8], agreed by the community and equivalent to some of them.

Illustration 23 shows the individuals of the ontology *licenses*, that contains the individuals of the subclasses of the class *License*, that is, *Apache*, *BSD*, etc. The individuals linked with *sameAs* have been signaled.

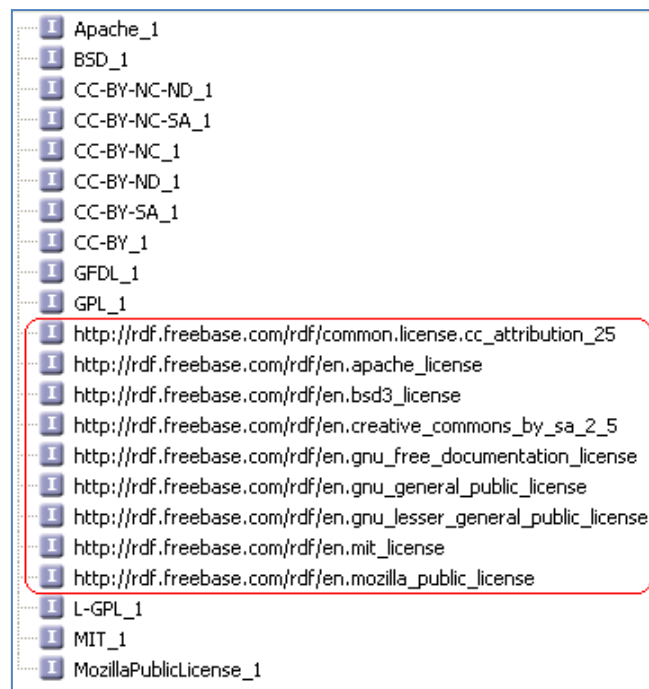


Illustration 23 – Individuals of the ontology licenses

Illustration 24 shows the individuals of the ontology *conditions*, that contains the individuals of the subclasses of the class *SpecialCondition*.

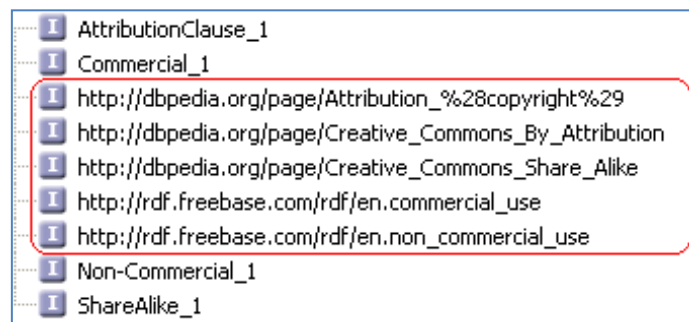


Illustration 24 – Individuals of the ontology conditions

Illustration 25 shows the individuals of the ontology *rights*, that contains the individuals of the subclasses of the class *Right*.

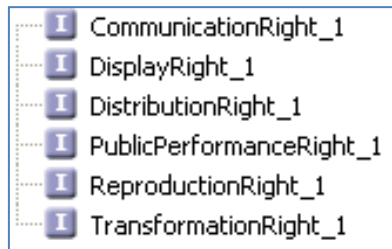


Illustration 25 – Individuals of the ontology rights

Illustration 26 shows the relationships of the individual *CC-BY-NC-ND_1*.

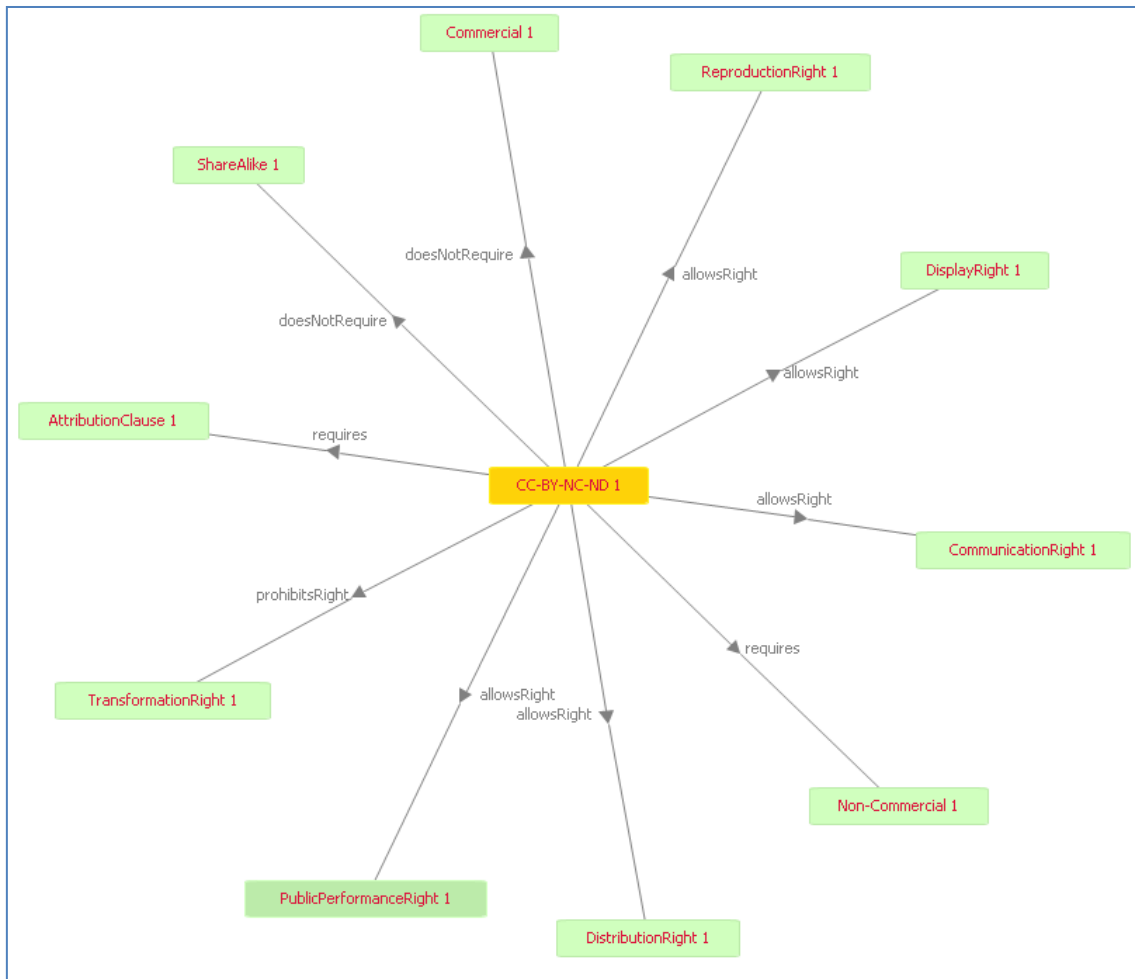


Illustration 26 – Relationships of the individual CC-BY-NC-ND_1

It can be observed that this license (the most restrictive *Creative Commons* license) allows of the rights (taken into account in the ontology) but the transformation right (*No Derivative*) and assuming no commercial uses (*Non-Commercial*).

Illustration 27 shows Manchester syntax of the individual *CC-BY-NC-ND_1* and shows *textually* that it also inherits from classes *CC-BY-NC-ND* and *Thing*.

▼ Frame Representation in Manchester Syntax

Individual: basic:CC-BY-NC-ND_1

Types:
 basic:CC-BY-NC-ND,
 owl:Thing

Facts:
 basic:allowsRight basic:CommunicationRight_1,
 basic:allowsRight basic:DisplayRight_1,
 basic:allowsRight basic:DistributionRight_1,
 basic:allowsRight basic:PublicPerformanceRight_1,
 basic:allowsRight basic:ReproductionRight_1,
 basic:doesNotRequire basic:Commercial_1,
 basic:doesNotRequire basic:ShareAlike_1,
 basic:prohibitsRight basic:TransformationRight_1,
 basic:requires basic:AttributionClause_1,
 basic:requires basic:Non-Commercial_1

Illustration 27 –Manchester syntax of the individual CC-BY-NC-ND_1

OWL code of the individual *CC-BY-NC-ND_1* is shown in Illustration 28 (properties *doesNotRequire* y *prohibitsRight* are excluded to increase legibility).

```

<owl:Thing rdf:about="#CC-BY-NC-ND_1">
  <rdf:type rdf:resource="#CC-BY-NC-ND_1"/>
  <requires rdf:resource="#AttributionClause_1"/>
  <doesNotRequire rdf:resource="#Commercial_1"/>
  <allowsRight rdf:resource="#CommunicationRight_1"/>
  <allowsRight rdf:resource="#DisplayRight_1"/>
  <allowsRight rdf:resource="#DistributionRight_1"/>
  <requires rdf:resource="#Non-Commercial_1"/>
  <allowsRight rdf:resource="#PublicPerformanceRight_1"/>
  <allowsRight rdf:resource="#ReproductionRight_1"/>
  <doesNotRequire rdf:resource="#ShareAlike_1"/>
  <prohibitsRight rdf:resource="#TransformationRight_1"/>
</owl:Thing>
    
```

Illustration 28 – OWL code of the individual CC-BY-NC-ND_1

Illustration 29 shows the Basic ontology statistics.

Category	Count
Classes	36
Data Properties	19
Object Properties	17
Annotation Properties	12
Datatypes	4
Individuals	31
Axioms	463
Logical Axioms	289
Subclass Axioms	30
Imported Ontologies (direct)	6
Imported Ontologies (closure)	8

Illustration 29 – Basic Ontology statistics

As can be shown, it is a small Ontology with 36 classes, 19 data properties, 17 object properties and 31 individuals.

Lastly, it is worth mentioning that this ontology has been evaluated by the *RaDON* plug-in, stating that it is a coherent a consistent ontology.

Property	Value
Ontology physical URI	file:/C:/Documents%20and%20Settings/Miguel/Escritorio/NeOnToolkit/workspace/basic5C/basic.owl
Ontology logical URI	http://www.semanticweb.org/ontologies/2009/10/11/basic.owl
All axioms	325
Coherent	true
Consistent	true
Unsatisfiable classes	0

Buttons: Handle Incoherence, Handle Inconsistency, Reload Ontology

Illustration 30 – Basic Ontology evaluation using *RaDON* plug-in

References

- [1] <http://www.mindswap.org/2003/owl/foaf>
- [2] <http://ontologies.semanticarts.com/dublincore/dublincore.owl>
- [3] http://purl.org/NET/dc_owl2dl/terms_od
- [4] Neon Toolkit Wiki. http://neon-toolkit.org/wiki/Main_Page
- [5] Suárez–Figuroa, M. C., Gómez–Pérez, A., and Villazón–Terrazas, B. 2009. How to Write and Use the Ontology Requirements Specification Document. Lecture Notes In Computer Science, vol. 5871. Springer–Verlag, Berlin, Heidelberg, 966–982.
- [6] Semantic Copyright. <http://semanticcopyright.org/>
- [7] DBpedia. <http://dbpedia.org/About>
- [8] Freebase. <http://www.freebase.com/>