LIPN Annotator manual

Ehab HASSAN, Sylvie SZULMAN, Sylvie SALOTTI
(Paris 13 - LIPN/RCLN)

1 Introduction

This document is the user guide of ***Annotator***.

Annotator is an autonomous java Eclipse RCP application with a graphical interface used to annotate a text with respect to a given ontology OWL and thesaurus SKOS and allows to link textual elemnts to conceptual resources. The annotation consists to mark text units corresponding to the elements of the ontology (Concept, Instance, Relation) or of The thesaurus which contains terminology units associated with the elemnts of the ontology.

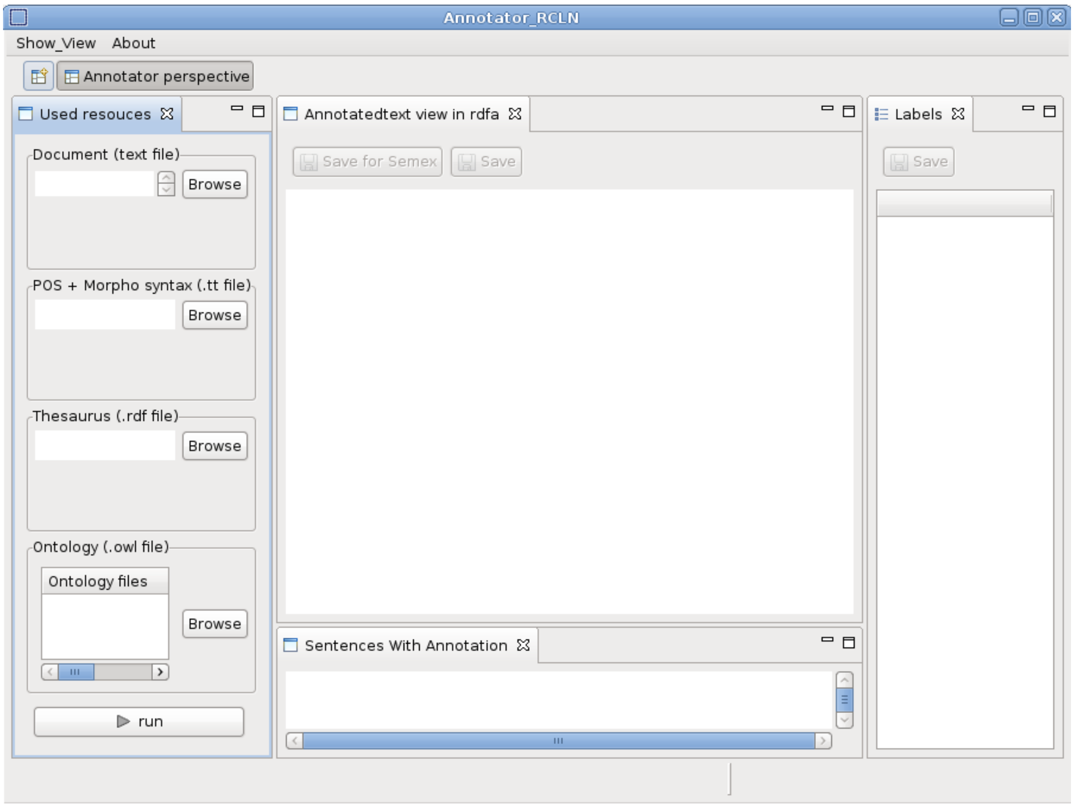


FIGURE 1 – Annotator overall interface

The present document describes the functionalities of the Annotator platform. The following sections present its main perspective and their functionalities.

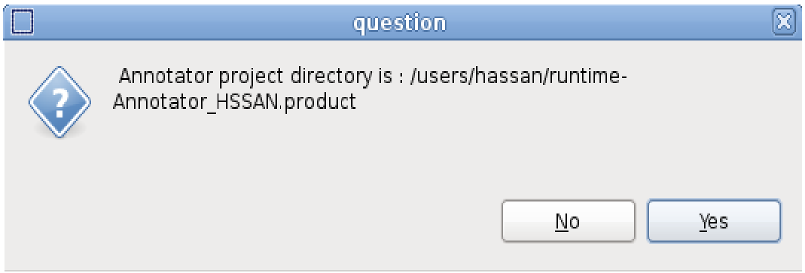
1.1 Installation

To install Annotator, you need java, version 3.6. It is provided for Linux, Mac OS X and Windows. Choose the convenient compressed version on the site [1], download it and uncompress it where you keep applicaions.

[1] http ://www-lipn.univ-paris13.fr/ szulman/Annotator/annotator.html

To lanuch it, double-click in the newly created directory on the icon of the application.

When the Annotator is launched, choose the directory of your project (if you don't agree the default value, a browser will ask for another one. see Figure.2) and the input encoding (if you don't agree UTF-8, you'll have to choose another one). The output will be encoded in UTF-8.



Annotator project directory is : /users/hassan/runtime-Annotator_HSSAN.product

FIGURE 2 – Dialogue window : select the directory

Then a window opens (see Figure.1) with four fields in the left pane *("Used resources")* and with a blank middle pane entitled *("Annotated text view")*, a blank right pane entitled *("Labels")* and a blank bottom pane entitled *("Sentences With Annotation")*.
Browse in the four left pane fields for the files which have been prepared.
Then run by clicking on the button with a triangle down this pane.

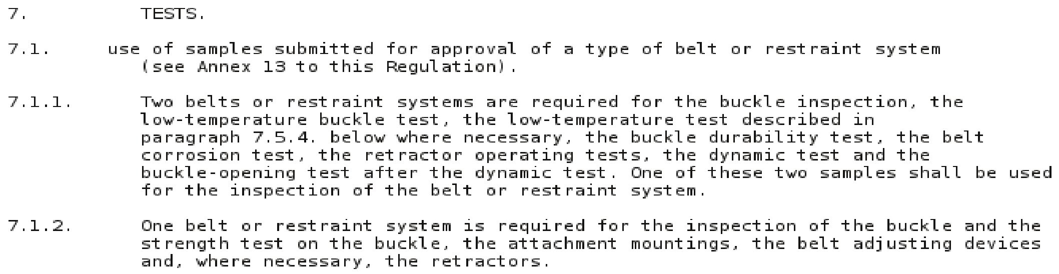
The annotated text appears in the pane *("Annotated text view")*, which will move to the left because the pane *("Used resources")* will disappear, You can check it and, if satisfied, save it : two buttons up the left pane allow to save either a project with the platform *SemEx* can use, or only two files describing the annotations according two different formats. Then, if you continue annotating some more files for *SemEx*, you can store the new results in the same project or create a fresh one.

2 General presentation of Annotator

2.1 Inputs/Outputs

To annotate a document, you need 4 inputs :

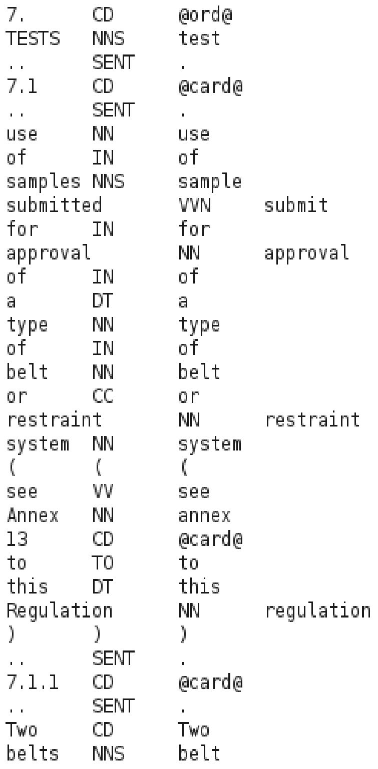
1. The document itself, in a single text (.txt) file (see Figure. 3) ;



```
7.          TESTS.

7.1.     use of samples submitted for approval of a type of belt or restraint system
            (see Annex 13 to this Regulation).

7.1.1.      Two belts or restraint systems are required for the buckle inspection, the
            low-temperature buckle test, the low-temperature test described in
            paragraph 7.5.4. below where necessary, the buckle durability test, the belt
            corrosion test, the retractor operating tests, the dynamic test and the
            buckle-opening test after the dynamic test. One of these two samples shall be used
            for the inspection of the belt or restraint system.

7.1.2.      One belt or restraint system is required for the inspection of the buckle and the
            strength test on the buckle, the attachment mountings, the belt adjusting devices
            and, where necessary, the retractors.
```

FIGURE 3 – Example of source document

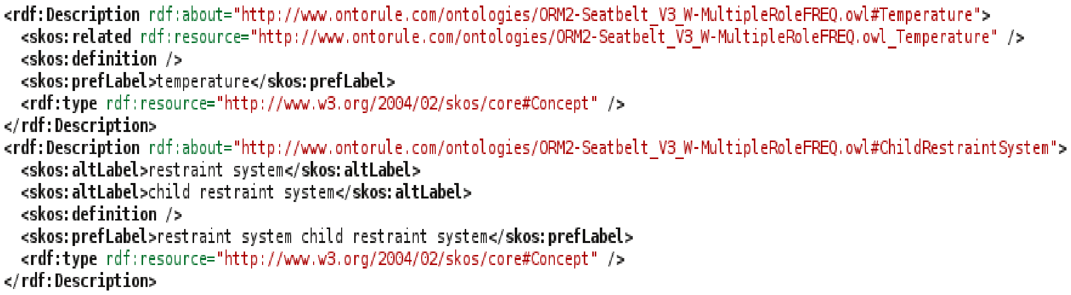
2. The output of a morphological analyzer and POS tagger, in three tabseperated columns (word, POS, lemma). we used *TreeTagger* (see Figure. 4) ;



```
7.      CD      @ord@
TESTS   NNS     test
..      SENT    .
7.1     CD      @card@
..      SENT    .
use     NN      use
of      IN      of
samples NNS     sample
submitted       VVN     submit
for     IN      for
approval        NN      approval
of      IN      of
a       DT      a
type    NN      type
of      IN      of
belt    NN      belt
or      CC      or
restraint       NN      restraint
system  NN      system
(       (       (
see     VV      see
Annex   NN      annex
13      CD      @card@
to      TO      to
this    DT      this
Regulation      NN      regulation
)       )       )
..      SENT    .
7.1.1   CD      @card@
..      SENT    .
Two     CD      Two
belts   NNS     belt
```

FIGURE 4 – Example of TreeTagger input

3. A lexicalization file following the SKOS standard (see Figure. 5) ;
4. an ontology in OWL format .

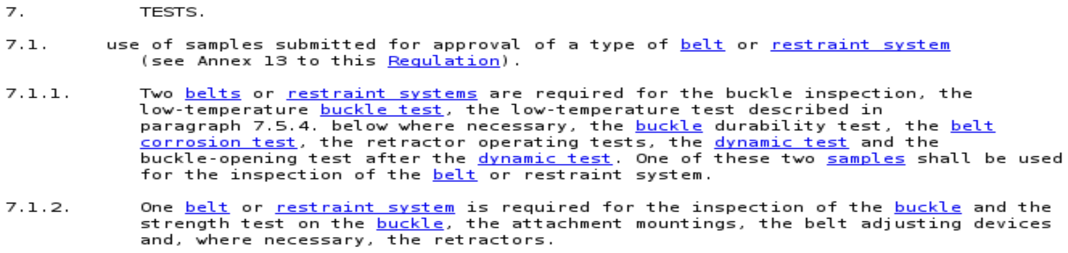


```
<rdf:Description rdf:about="http://www.ontorule.com/ontologies/ORM2-Seatbelt_V3_W-MultipleRoleFREQ.owl#Temperature">
  <skos:related rdf:resource="http://www.ontorule.com/ontologies/ORM2-Seatbelt_V3_W-MultipleRoleFREQ.owl_Temperature" />
  <skos:definition />
  <skos:prefLabel>temperature</skos:prefLabel>
  <rdf:type rdf:resource="http://www.w3.org/2004/02/skos/core#Concept" />
</rdf:Description>
<rdf:Description rdf:about="http://www.ontorule.com/ontologies/ORM2-Seatbelt_V3_W-MultipleRoleFREQ.owl#ChildRestraintSystem">
  <skos:altLabel>restraint system</skos:altLabel>
  <skos:altLabel>child restraint system</skos:altLabel>
  <skos:definition />
  <skos:prefLabel>restraint system child restraint system</skos:prefLabel>
  <rdf:type rdf:resource="http://www.w3.org/2004/02/skos/core#Concept" />
</rdf:Description>
```

FIGURE 5 – Example of SKOS input

Annotator outputs :

1. The input document annotated with respect to the semantic ontological elements (RDFa format, see Figure. 6) ;



```
7.          TESTS.

7.1.     use of samples submitted for approval of a type of belt or restraint system
            (see Annex 13 to this Regulation).

7.1.1.      Two belts or restraint systems are required for the buckle inspection, the
            low-temperature buckle test, the low-temperature test described in
            paragraph 7.5.4. below where necessary, the buckle durability test, the belt
            corrosion test, the retractor operating tests, the dynamic test and the
            buckle-opening test after the dynamic test. One of these two samples shall be used
            for the inspection of the belt or restraint system.

7.1.2.      One belt or restraint system is required for the inspection of the buckle and the
            strength test on the buckle, the attachment mountings, the belt adjusting devices
            and, where necessary, the retractors.
```

FIGURE 6 – Example of RDFa annotation. The blue text corresponds to the annotated term

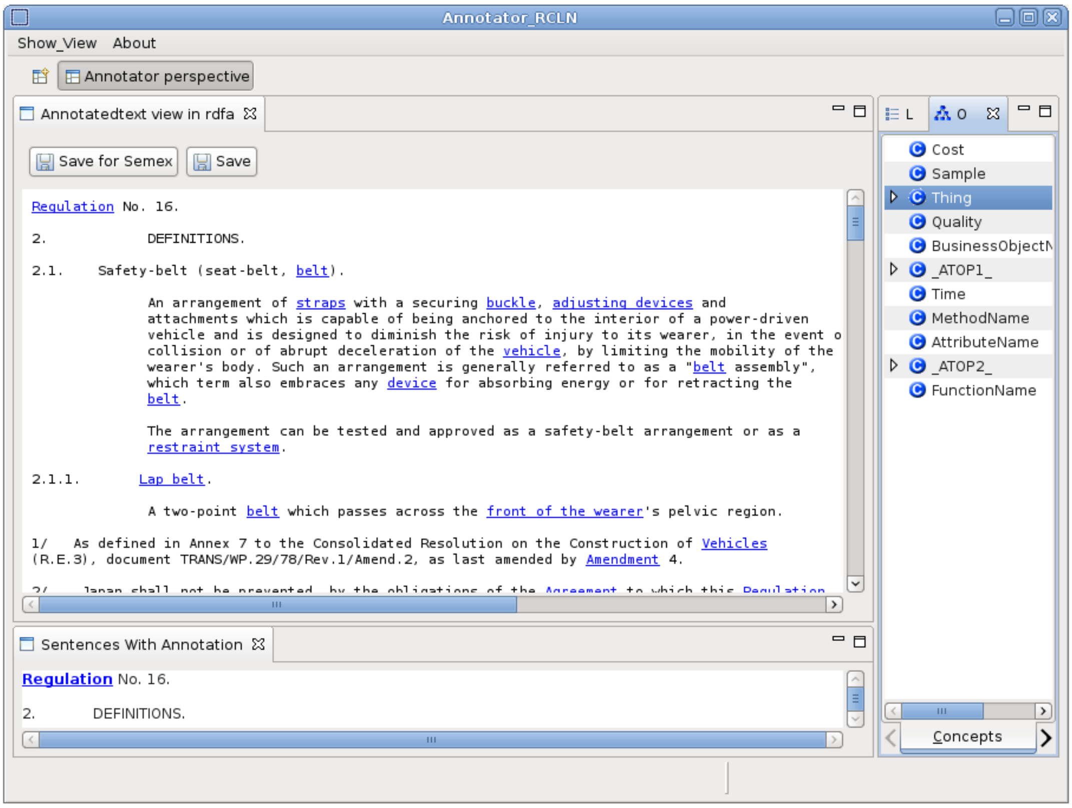
2. a HTML file *"rdfaOutput.html"* with the annotated document thats allow to see the result in the web.
3. a text file *"textAnnote.txt"* that contains all the annotated sentences with their annotations.
4. a file XML *"corpusAnnot2XML.xml"* that contains all the sentences of text with the annotated words and some information about each word (Position in the sentence, The Target,..) ;

```
<Sentence xmlns="" ID="doc-0:sent-2">
      <content_sent>Safety-belt (seat-belt, belt).</content_sent>
      <start_sent>60</start_sent>
      <end_sent>89</end_sent>
      <Annotation>
         <content>belt</content>
         <start_offset>24</start_offset>
         <end_offset>27</end_offset>
         <target>"http://www.ontorule.com/ontologies/
            ORM2-Seatbelt_V3_W-MultipleRoleFREQ.owl#SeatBelt"</target>
         <type_Resource>resource not exist</type_Resource>
      </Annotation>
   </Sentence>
 ........
```

5. The ontological hierarchy *(Concept hierarchy, Property hierarchy).*

2.2 Presentation of the perspective

The Annotator perspective enables to annotate a text, either a large document or a small fragement of text, with respect to a given ontology and thesaurus. This perspective is composed of four main windows allows to navigate through a document, through an ontology *(concepts, instances and properties)* (see Figure. 7). but you can modify this presentation using the minimize or maximize buttons associated to each window :



```
Regulation No. 16.

2.            DEFINITIONS.

2.1.    Safety-belt (seat-belt, belt).

              An arrangement of straps with a securing buckle, adjusting devices and
              attachments which is capable of being anchored to the interior of a power-driven
              vehicle and is designed to diminish the risk of injury to its wearer, in the event o
              collision or of abrupt deceleration of the vehicle, by limiting the mobility of the
              wearer's body. Such an arrangement is generally referred to as a "belt assembly",
              which term also embraces any device for absorbing energy or for retracting the
              belt.

              The arrangement can be tested and approved as a safety-belt arrangement or as a
              restraint system.

2.1.1.       Lap belt.

              A two-point belt which passes across the front of the wearer's pelvic region.

1/   As defined in Annex 7 to the Consolidated Resolution on the Construction of Vehicles
(R.E.3), document TRANS/WP.29/78/Rev.1/Amend.2, as last amended by Amendment 4.

2/    Japan shall not be prevented  by the obligations of the Agreement to which this Regulation
```

Regulation No. 16.

2. DEFINITIONS.

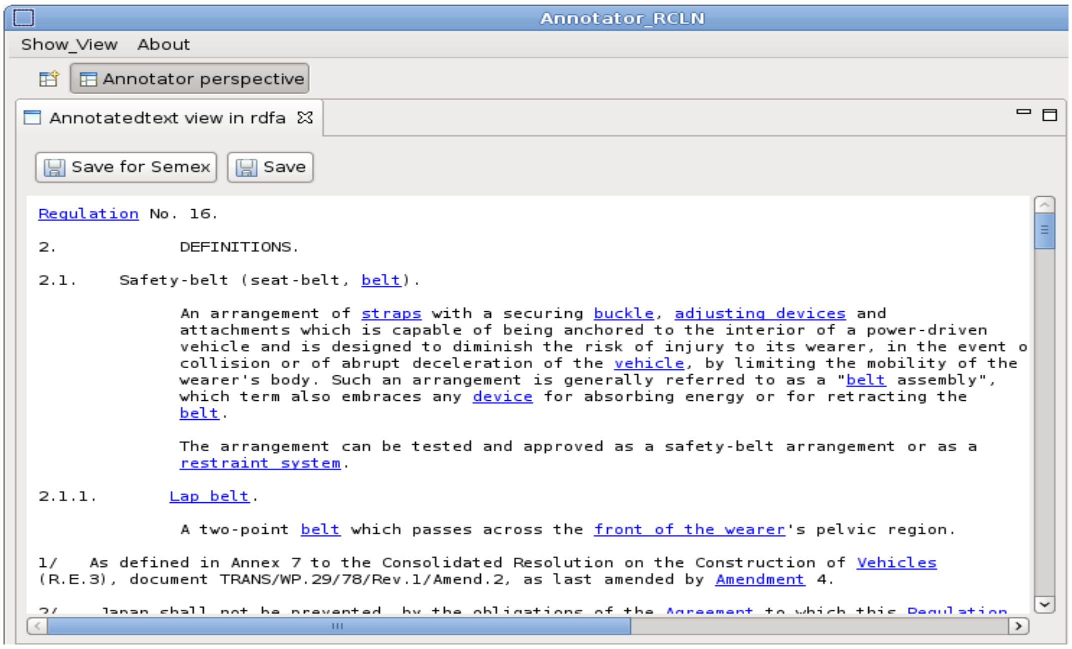
FIGURE 7 – Annotator result interface

Used resource window : in this window we have four fields allow to select the four input files required by clicking in the button *"Browse"*.
Once this is done, the relevant data are loaded and you are ready to work by clicking in the *"run"* button in the bottom used to get statrted the annotation (see Figure. 1).

Annotated text window : This window presents the annotated corpus as a *RDFa* document in which the term that are annotated with terminology entities are emphasized and anchored using *HTML* link (see Figure. 8). By clicking on the button save, you can save the result in a *HTML* file to view the annotated text in *RDFa* in a browser.

Labels window : The right part of the Annotator perspective is used to visualise the occurrence of the terms.

This perspective enables the users to link the term with the sentence where this word has been found and to see some informations about each term.



Regulation No. 16.

2. DEFINITIONS.

2.1. Safety-belt (seat-belt, belt).

An arrangement of straps with a securing buckle, adjusting devices and attachments which is capable of being anchored to the interior of a power-driven vehicle and is designed to diminish the risk of injury to its wearer, in the event o collision or of abrupt deceleration of the vehicle, by limiting the mobility of the wearer's body. Such an arrangement is generally referred to as a "belt assembly", which term also embraces any device for absorbing energy or for retracting the belt.

The arrangement can be tested and approved as a safety-belt arrangement or as a restraint system.

2.1.1. Lap belt.

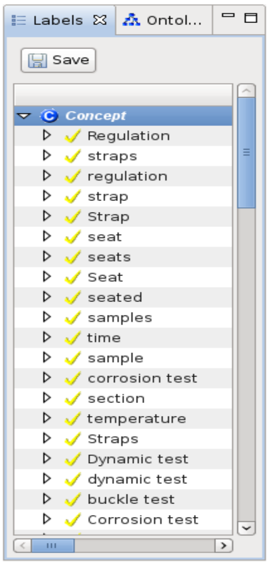
A two-point belt which passes across the front of the wearer's pelvic region.

1/ As defined in Annex 7 to the Consolidated Resolution on the Construction of Vehicles (R.E.3), document TRANS/WP.29/78/Rev.1/Amend.2, as last amended by Amendment 4.

2/ Japan shall not be prevented by the obligations of the Agreement to which this Regulation

FIGURE 8 – Annotated text window

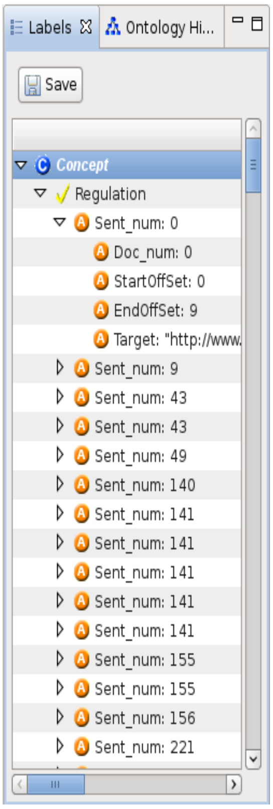
Here, we have four root, three root present the resource type *(Concept, Instance, Property)* extracted from the ontolgy. The fourth root is ***(Resource not exist)*** which indicates the absence of the resource from the ontology . Clicking on the black triangle to the left of any root allows to visualise or hide the list of terms thats have this resource type (see Figure. 9).



Concept
- ✓ Regulation
- ✓ straps
- ✓ regulation
- ✓ strap
- ✓ Strap
- ✓ seat
- ✓ seats
- ✓ Seat
- ✓ seated
- ✓ samples
- ✓ time
- ✓ sample
- ✓ corrosion test
- ✓ section
- ✓ temperature
- ✓ Straps
- ✓ Dynamic test
- ✓ dynamic test
- ✓ buckle test
- ✓ Corrosion test

FIGURE 9 – the termes that have the type source "Concept"

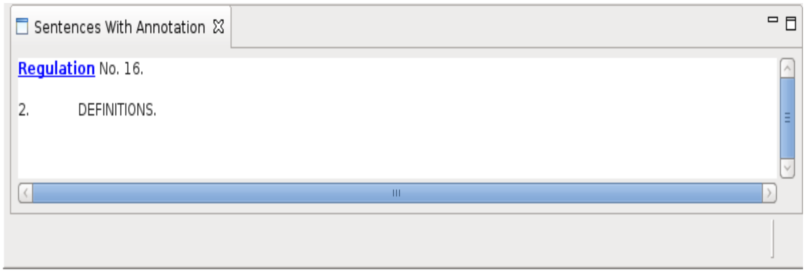
as well as, if you click on the black triangle to the left of any term, you will see the numbers of sentences which have this term. we can visulaise more information about any occurrence of the term such as (the number of document, la position of the occurrence in the sentence and the target that presents the uri extracted from the SKOS file) by clicking on the black triangle to the left of the occurrence (see Figure. 10). one button up this pane allow to save this result as a file XML.



```
▽ Concept
  ▽ Regulation
    ▽ Sent_num: 0
        Doc_num: 0
        StartOffSet: 0
        EndOffSet: 9
        Target: "http://www.
    ▷ Sent_num: 9
    ▷ Sent_num: 43
    ▷ Sent_num: 43
    ▷ Sent_num: 49
    ▷ Sent_num: 140
    ▷ Sent_num: 141
    ▷ Sent_num: 141
    ▷ Sent_num: 141
    ▷ Sent_num: 141
    ▷ Sent_num: 141
    ▷ Sent_num: 155
    ▷ Sent_num: 155
    ▷ Sent_num: 156
    ▷ Sent_num: 221
```

FIGURE 10 – the terms occurrences with the additional information

Sentences With Annotation window : The bottom part of the Annotator perspective present the sentence that have the annotated word (see Figur. 11). you can visulaise the sentences in this part by clicking on the sentence number which exists in the previous window.

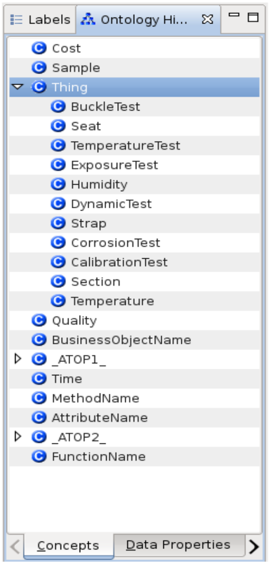


Regulation No. 16.

2. DEFINITIONS.

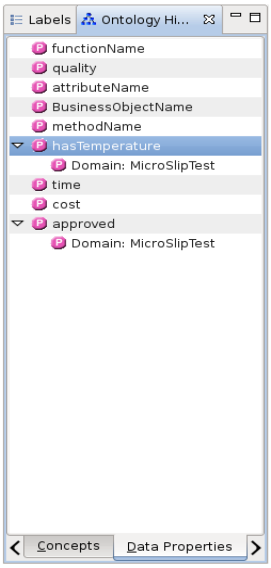
FIGURE 11 – the annotated sentences

Ontology Hierarchy window : The ontological hierarchy is presented on the right of the perspestive. Three tabs at the bottom of the window allow to visualize the concept hierarchy (see Figure. 12), the data property hierarchy (see Figure. 13) or the objects property hierarchy (see Figure. 14). Clicking on the black triangle to the left a concept or property allows to visualize or hide the sub-hierarchy of that concept or property.



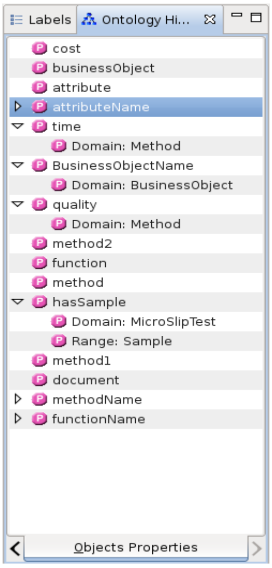
- Cost
- Sample
- Thing
  - BuckleTest
  - Seat
  - TemperatureTest
  - ExposureTest
  - Humidity
  - DynamicTest
  - Strap
  - CorrosionTest
  - CalibrationTest
  - Section
  - Temperature
- Quality
- BusinessObjectName
- _ATOP1_
- Time
- MethodName
- AttributeName
- _ATOP2_
- FunctionName

FIGURE 12 – concept hierarchy



- functionName
- quality
- attributeName
- BusinessObjectName
- methodName
- hasTemperature
  - Domain: MicroSlipTest
- time
- cost
- approved
  - Domain: MicroSlipTest

FIGURE 13 – data property hierarchy



- cost
- businessObject
- attribute
- attributeName
- time
  - Domain: Method
- BusinessObjectName
  - Domain: BusinessObject
- quality
  - Domain: Method
- method2
- function
- method
- hasSample
  - Domain: MicroSlipTest
  - Range: Sample
- method1
- document
- methodName
- functionName

FIGURE 14 – objects property hierarchy

3 Program

as soon as the user enter the 4 input file and clicking on the button *"Run"*, several tasks will be performed for annotate the text :

3.1 cut the source document into sentences

we used morphological analyzer *"TreeTagger"* which lets us know the end of the sentences by the word "SEND" in the columns POS. for well cuting the text in sentences and identifying the position of each sentence in the text, we read the file (.tt), that contains the three tabseperated columns *(word, POS, lemma)*, line by line, until the end of the sentence and put all the words and lemmas in lowercase (The lemma *"unkown"* are replaced by the word). Then we run throught the orginal text by counting only the caracters considered by TreeTagger and put them in a list with the position and the number of each sentence.

3.2 Looking for the occurrences in the text

for this task, we read the lexicalization file (SKOS) to extract the terminological units *(Concept, prefLabel, altLabel)*. Then, we extract the list of lemma corresponding to the term by comparing this term with the word or the lemma of TreeTagger. after that, we search all the occurrences of the terms in the text by comparing the first word of the term with the lemmas. if there is a match, we extract the correspondce word and we compare the second word with the next lemma and so on. for each occurrence found, we calculate the offset in the sentence.

At the end of this step, we will have for each concept a dictionary which thier keys are the terms and thier values are the occurrences lists which include les occurrences of the terms with additionals informations extracted from the sentence like the position of the occurrence, the sentence number. In addition, there will be a XML output file containig all the sentences with their positions in the orginal text.

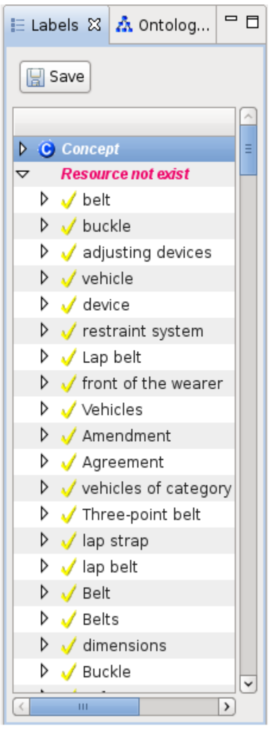
3.3 Text Annotation

after axtracting the occurrences, we take care to annotate the text sentence by sentence. for this :

- we get the resource type *(Concept, Instance, Relation)* from the ontology by :
  1. getting the Label Entity from the SKOS file for each occurrence, e.g. the word *#Regulation* from the concept *"http :// www.ontorule.com/ontologies/ORM2-Seatbelt_ V3_ W-MultipleRoleFREQ.owl#Regulation"*.
  2. getting the URI from the ontology, e.g. *"http ://www.ontorule.com/ontologies/ORM2-Seatbelt_ V3_ W-MultipleRoleFREQ.owl"*.
  3. adding this URI to each Label Entity, e.g. we have *"http ://www.ontorule.com/ontologies /ORM2-Seatbelt_V3_W-MultipleRoleFREQ.owl#Regulation"*.
  4. then, we look for this concept in the ontology to find the resource type.

for the concepts which do not exist in the OWL file, there are two possibility :

1. we annotate the sentence with the resource type ***"Resource not exist"***, i.e. this concept appears in the SKOS file. but, it does not exists in the ontology. in this case, we can visualize this occurrences in the *"Labels window"* under the root *"Resource not exist"* (see Figure. 15).



Concept
Resource not exist
✓ belt
✓ buckle
✓ adjusting devices
✓ vehicle
✓ device
✓ restraint system
✓ Lap belt
✓ front of the wearer
✓ Vehicles
✓ Amendment
✓ Agreement
✓ vehicles of category
✓ Three-point belt
✓ lap strap
✓ lap belt
✓ Belt
✓ Belts
✓ dimensions
✓ Buckle

FIGURE 15 – the termes that have the type source "resource not exist"

2. we do not consider this occurrences, i.e. we do not make the annotation for them.

– for each occurrence exists in the sentence, we add the annotations which contains the four variables :

Start_ offset : the beginning of the occurrence.
End_ offset : the end of the the occurrence.
prefix : this variable should be added before the occurrence in the sentence.

prefix = "<span about="elemName+(number)" typeof="schema :elemType" rel="schema :realize"elemType">" + href;

where :

href = "<a class = elemType about = elemName href="javascript :void() ; "onclick="function('shortName') ;">".
elemType : The resource type extract from the ontology, e.i. Concept, Relation, Instance.
elemName : URI + Label Entity.
shortName : the occurrence.

suffix : we add this variable after the occurrence.

suffix="</a></span>".

exemple :

<span about="http ://www.ontorule.com/ontologies/ORM2-Seatbelt_V3_W- MultipleRoleFREQ.owl#Regulation0" typeof="schema :Concept" rel="schema :realizeConcept" ><a class="Concept" about="http ://www.ontorule.com/ontologies /ORM2-Seatbelt_V3_W-MultipleRoleFREQ.owl# Regulation" href="javascript :void() ;"onclick="function('Regulation') ;" >Regulation </a></span>

Here, for a given occurrence, we can have more than one annotation due to the frequency of labels *(prefLabel, altLabel)* in the Skos file. for that, we consider the longest annotation and remove the anothers.

- Then, for each sentence which have an occurrence, we add another *prefix1* in the beginng of the the sentence and *suffix1* in the end of this sentence.

  prefix1 = "<span id="textlink"+numlink+ typeof="schema :TextLink"rel="schema :defineResource">"

  suffix1 = "</span>".

- after that, we insert a new *prefix2* after *prefix1* and new *suffix2* after *suffix1* for all the sentences :

  prefix2 = "<span id="+sentid" about="namespace+sentid typeof="schema :Sentence" rel="schema :annoted" class="Sentence">".

  where :

  sentid = the sentence number.
  namespace = "http ://lipn.univ-paris13.fr/RCLN/SemEx/text#".
  suffix2 = "</span>".

  exemple :

  <span id="sent0-0" about="http ://lipn.univ-paris13.fr/RCLN/SemEx/text# sent0-0" typeof="schema :Sentence" rel="schema :annoted" class="Sentence">

- finally, we add the *HTML* prefix and *HTML* suffix to have the ability to display the annotated text by web Browser.

  htmlPREFIX = "< ?xml version='1.0' encoding="UTF-8" standalone="no" ?>
  <html xmlns="http ://www.w3.org/1999/xhtml"
  version="XHTML+RDFa 1.0"
  xmlns :dc="http ://purl.org/dc/elements/1.1/" xmlns :rdf="http ://www.w3.org/1999/02/22-

```
rdf-syntax-ns# "
xmlns :schema="http ://lipn.univ-paris13.fr/RCLN/schema# "
>
<head>
<meta content="text/html ; charset=UTF-8 http-equiv="content-type"/>
<link rel="stylesheet" type="text/css" href="cssStyle.css"/>
<script language="JavaScript" src="script.js">
</script>
</head>
<body>
<pre>"
htmlSUFFIX = "</pre> </body> </html>" ;
```

- in addition, we have built a text file *"textAnnote"* containing all the sentences with all the annotations for each occurrence.