Controlled Natural Language rendering of Copyright Ontology licenses

Roberto García, Rosa Gil

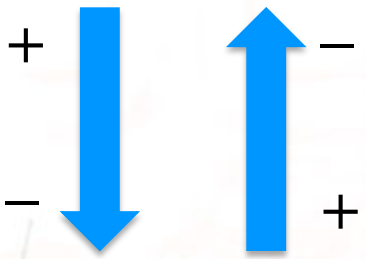
HCI & Data Integration
Research Group
Universitat de Lleida, Spain

Table of Contents

- Introduction
- Motivation
- Proposal
- Copyright Model
- License Modelling
- License Rendering
- Conclusions
- Future Work

Introduction

- In general, RELS seek **more expressivity**… but **trade-off**:
  - **Formality**:
    - more sophisticated computerised processing



  - **Usability**:
    - more easily understandable by end-users

Introduction

- **Creative Commons**:
  - very usable but very limited expressiveness
  - **CC+**: informal extensions
- **Logic-based** RELs:
  - very expressive (formal languages) but difficult for end-users
- …**XML-based** RELs:
  - tradeoff…
    as long as XML-aware users

Motivation

- Is it possible a **formal REL** that **renders nice** to end-users?
  - Formal semantics (ontology)
  - Automated reasoning
  - Natural language rendering
  - …
- Our attempt:
  - the **Copyright Ontology**

Proposal

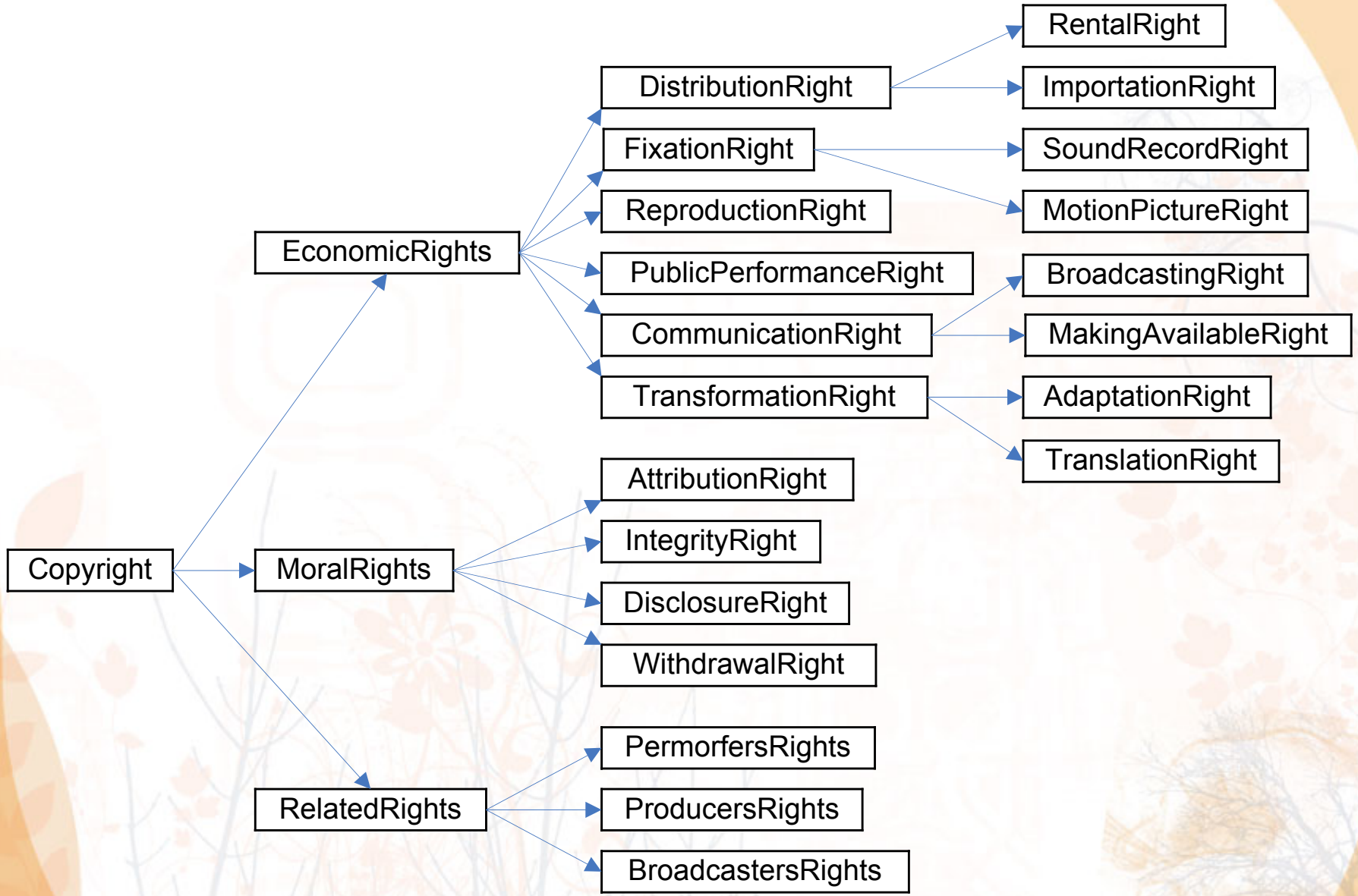
- **Copyright Ontology** [1] features:
  - More **expressive** than XML approaches
  - Facilitates **interoperability**, works at the semantic level
  - Based on a **copyright law model**, WIPO worldwide recommendations [2]
  - Also based on **Semantic Web** technologies

- Copyright Ontology:
  provides the **building blocks** to flexibly **model licenses**…and **reason** about them

[1] Copyright Ontology, http://rhizomik.net/ontologies/copyrightonto
[2] World Intellectual Property Organisation, http://www.wipo.int

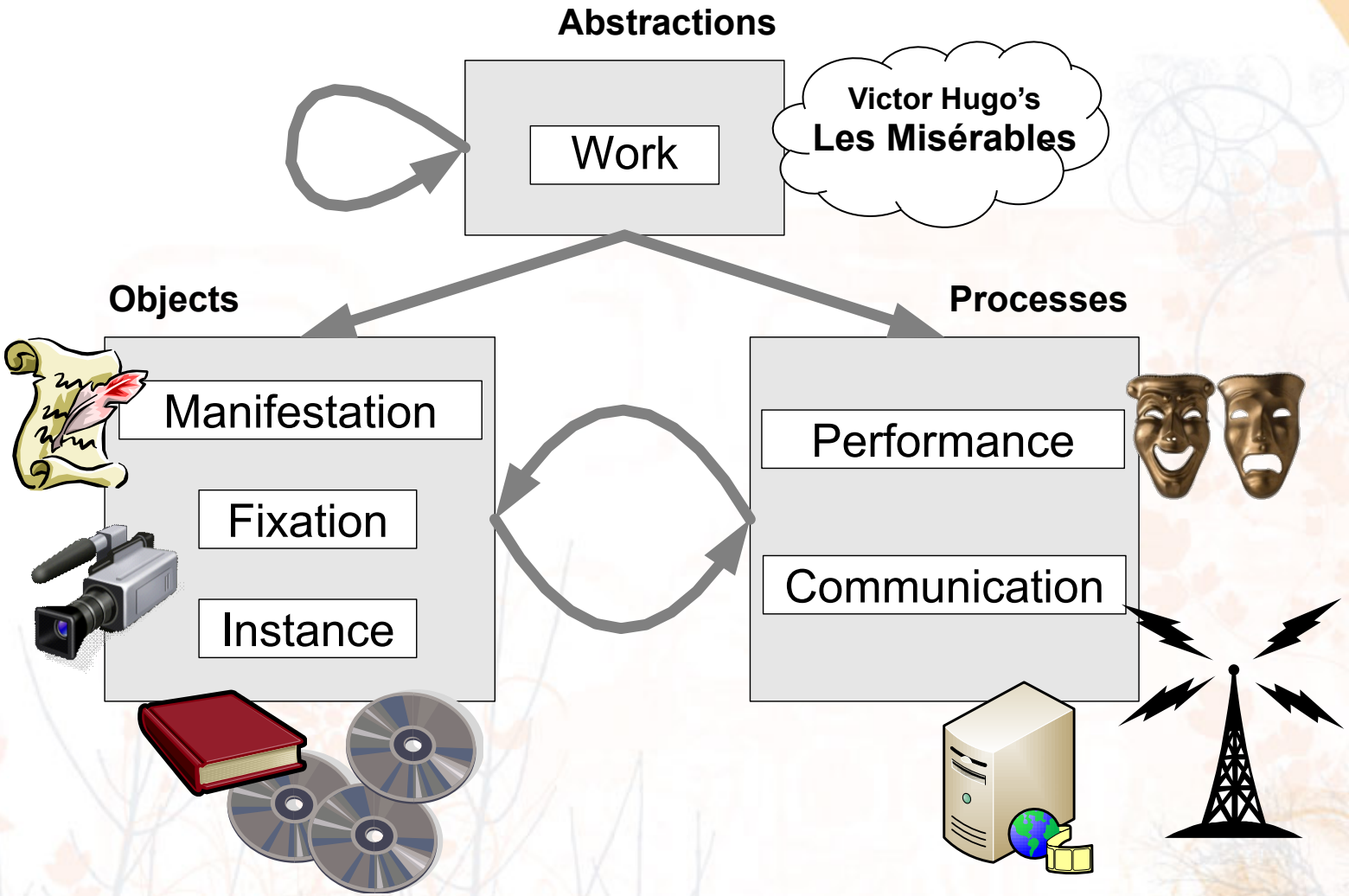
Copyright Model

Rights Model



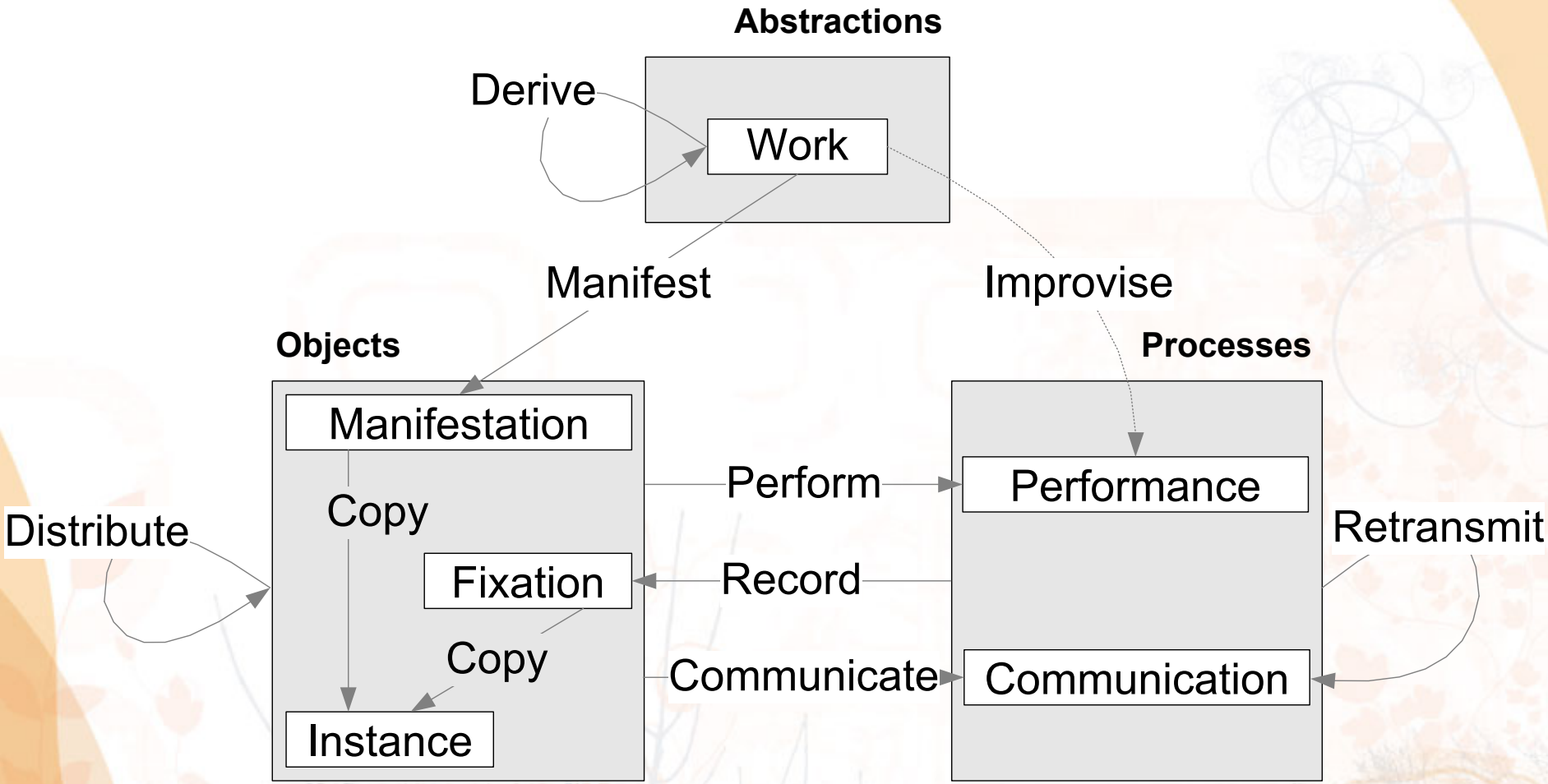
Copyright Model

Creation Model



Copyright Model

Action Model



Copyright Model

Action Model

- Actions governed by **economic rights**:

Right	Action
Reproduction	**Copy**
Distribution	**Distribute** (Sell, Rent, Lend)
Public Performance	**Perform**
Fixation	**Record**
Communication	**Broadcast, Make Available**
Transformation	**Derive** (Adapt, Translate)

License Modelling

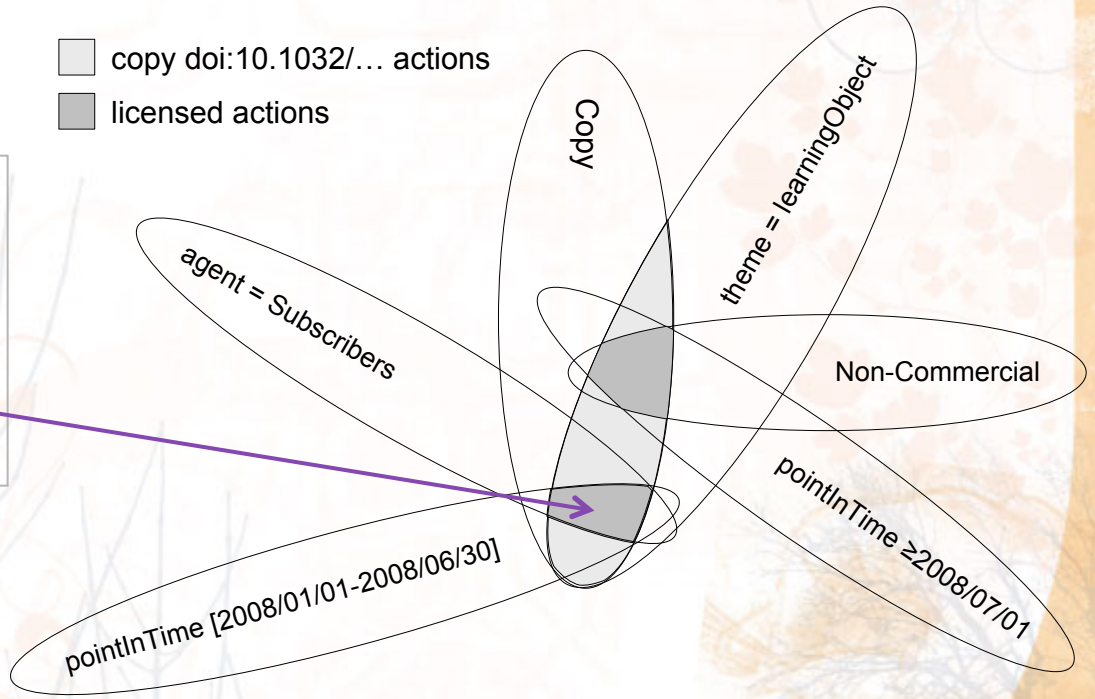
Actions –**case roles**→ Action Participants

	initiator	**resource**	**goal**	**essence**
Action	agent, effector	instrument	result, recipient	patient, theme
Process	agent, origin	matter	result, recipient	patient, theme
Transfer	agent, origin	instrument, medium	experiencer, recipient	theme
Spatial	origin	path	destination	location
Temporal	start	duration	completion	pointInTime
Ambient	reason	manner	aim, consequence	condition

License Modelling

- Combine these building blocks to **model licenses**
- Main component: class **Action Pattern**
  - Defined combining restrictions on case roles using logical operators

$$\begin{aligned} \text{Pattern} \equiv\ & \text{Copy} \sqcap \\ & \exists \text{pointInTime}.\geq 2008\text{-}01\text{-}01, \\ & \qquad\qquad \leq 2008\text{-}06\text{-}30 \sqcap \\ & \exists \text{agent}.\text{Subscribers} \sqcap \\ & \exists \text{theme}.\{\text{learningObject}\} \end{aligned}$$



License Modelling

- Additional actions and case roles:
  - **Agree** (Disagree): state what is permitted (prohibited)

```
:agreement a co:Agree;
  co:agent :owner;
  co:theme :Pattern.
```

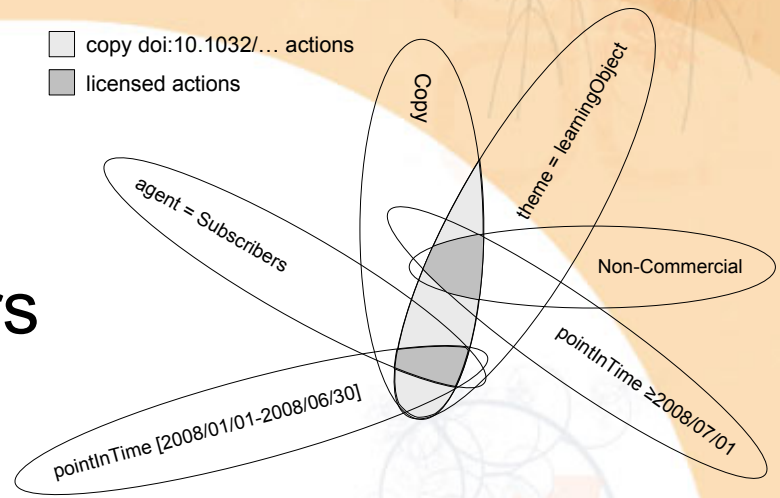
  - **condition**: links to another pattern that must satisfied
  - **aim**: links condition pattern to conditioned one

```
:Pattern
  co:condition :Condition.
```

```
Condition ≡ Transfer ⊓
  ∃recipient.{owner} ⊓
  ∃theme.{3EurosAmount}
  ∃agent.Subscribers ⊓
  ∃aim.Pattern
```

  - **consequence**: links pattern to an obliged one

License Modelling



- Description Logic reasoners implement classification:
  - check instance action classified in class for pattern
- Semantic queries check action in agreed pattern and condition fulfilment:

```
ASK {
  ?agreement rdf:type co:Agree;
             co:theme ?pattern.
  :copy      rdf:type ?pattern;
             co:agent ?consumer;
             co:condition ?conditionPattern.
  ?condition rdf:type ?conditionPattern;
             co:agent ?consumer;
             co:aim :copy.}
```

License Rendering

RDF/XML serialisation

```
<owl:Class rdf:about="&patterns;Pattern_01”>
  <co:condition rdf:resource="&patterns;Condition_01"/>
  <rdfs:subClassOf rdf:resource="&co;Copy"/>
  <owl:equivalentClass>
   <owl:Class>
    <owl:intersectionOf rdf:parseType="Collection”>
     <owl:Restriction>
      <owl:someValuesFrom rdf:resource="&persons;Subscribers”/>
      </owl:someValuesFrom>
      <owl:onProperty rdf:resource="&co;agent"/>
     </owl:Restriction>
     <owl:Restriction>
      <owl:maxCardinality rdf:datatype="&xsd;int">1</owl:maxCardinality>
      <owl:onProperty rdf:resource="&co;agent"/>
     </owl:Restriction>
     <owl:Restriction>
      <owl:hasValue rdf:resource="&creations;learningObject_01”/>
      <owl:onProperty rdf:resource="&co;theme"/>
     </owl:Restriction>
     <owl:Restriction>
      <owl:allValuesFrom>
       <owl:DataRange>
        <owl:onDataRange rdf:resource="&xsd;date"/>
         <owl:minInclusive>2008-06-01</owl:minInclusive>
         <owl:maxExclusive>2009-01-01</owl:maxExclusive>
        </owl:DataRange>
      </owl:allValuesFrom>
      <owl:onProperty rdf:resource="&co;pointInTime"/>
     </owl:Restriction>
    </owl:intersectionOf>
   </owl:Class>
  </owl:equivalentClass>
</owl:Class>
```

```
<owl:Class rdf:about="&patterns;Condition_01”>
 <rdfs:subClassOf rdf:resource="&co;Transfer"/>
 <owl:equivalentClass>
  <owl:Class>
   <owl:intersectionOf rdf:parseType="Collection”>
    <owl:Restriction>
     <owl:someValuesFrom rdf:resource="&persons;Subscribers"/>
     <owl:onProperty rdf:resource="&co;agent"/>
    </owl:Restriction>
    <owl:Restriction>
     <owl:maxCardinality rdf:datatype="&xsd;int">1</owl:maxCardinality>
     <owl:onProperty rdf:resource="&co;agent"/>
    </owl:Restriction>
    <owl:Restriction>
     <owl:hasValue rdf:resource="&persons;owner”/>
     <owl:onProperty rdf:resource="&co;recipient"/>
    </owl:Restriction>
    <owl:Restriction>
     <owl:hasValuerdf:resource="&misc;Amount3Euros”/>
     <owl:onProperty rdf:resource="&co;theme"/>
    </owl:Restriction>
    <owl:Restriction>
     <owl:someValuesFrom rdf:resource="&patterns;Pattern_01"/>
     <owl:onProperty rdf:resource="&co;aim"/>
    </owl:Restriction>
   </owl:intersectionOf>
  </owl:Class>
 </owl:equivalentClass>
</owl:Class>

<co:Agree rdf:about="&actions;agreement_01”>
  <co:agent rdf:resource="&persons;owner"/>
  <co:theme rdf:resource="&patterns;Pattern_01"/>
</co:Agree>
```

License Rendering

RDF/N3 serialisation

```
:Pattern_01     a :Class;
        co:condition :Condition_01;
        rdfs:subClassOf co:Copy;
        :equivalentClass  [
           a :Class;
           :intersectionOf  (
           [
                 :onProperty co:agent;
                 :someValuesFrom :Subscribers ]
           [
                 :maxCardinality "1"^^xsd:int;
                 :onProperty co:agent ]
           [
                 :hasValue :learningObject_01;
                 :onProperty co:theme ]
           [
                 :allValuesFrom  [
                    a :DataRange;
                    owl:maxExclusive "2009-01-01"^^xsd:date;
                    owl:minInclusive "2008-06-01"^^xsd:date;
                    owl:onDataRange xsd:date ];
                 :onProperty co:pointInTime ] ) ] .
```

```
:Condition_01     a :Class;
     rdfs:subClassOf co:Transfer;
     :equivalentClass  [
        a :Class;
        :intersectionOf  (
        [
              :onProperty co:agent;
              :someValuesFrom :Subscribers ]
        [
              :maxCardinality "1"^^xsd:int;
              :onProperty co:agent ]
        [
              :hasValue :owner;
              :onProperty co:recipient ]
        [
              :hasValue :Amount3Euros;
              :onProperty co:theme ]
        [
              :onProperty co:aim;
              :someValuesFrom :Pattern_01 ] ) ] .

:agreement_01     a co:Agree;
     co:agent :owner;
     co:theme :Pattern_01 .
```

License Rendering

RDF to HTML+RDFa (http://rhizomik.net/redefer)

agreement 01 a **Agree**

agent	owner
theme	Pattern 01

Pattern 01 a **Class**

condition	Condition 01
equivalentClass	**Class**
intersectionOf	**Restriction**
onProperty	agent
someValuesFrom	Subscribers
Restriction	
maxCardinality	1
onProperty	agent
Restriction	
hasValue	learningObject 01
onProperty	theme
Restriction	
allValuesFrom	**DataRange**
maxExclusive	2009-01-01T00:00:00
minInclusive	2008-06-01T00:00:00
onDataRange	dateTime
onProperty	pointInTime
subClassOf	Copy

Condition 01 a **Class**

equivalentClass	**Class**
intersectionOf	**Restriction**
onProperty	agent
someValuesFrom	Subscribers
Restriction	
maxCardinality	1
onProperty	agent
Restriction	
hasValue	owner
onProperty	recipient
Restriction	
hasValue	Amount3Euros
onProperty	theme
Restriction	
onProperty	aim
someValuesFrom	Pattern 01
Restriction	
maxCardinality	1
onProperty	aim
subClassOf	Transfer

License Rendering

Controlled Natural Language

Case Role	Mapping Condition	CNL
agent		(subject)
effector		(subject)
experiencer	there is not agent/effector	(subject)
instrument	there is not agent/effector/ experiencer otherwise	(subject) with
theme	range is verb range is not verb	that (end of sentence) (object)
patient		(object)
matter		of
medium		by
pointInTime		at
location		at
path		through
duration		for

Case Role	Mapping Condition	CNL
origin		from
start		from
origin		from
destination		to
recipient		to
result		resulting
completion		until
reason		with reason
manner		with manner
aim		with aim
consequence		with consequence
condition		with condition

License Rendering

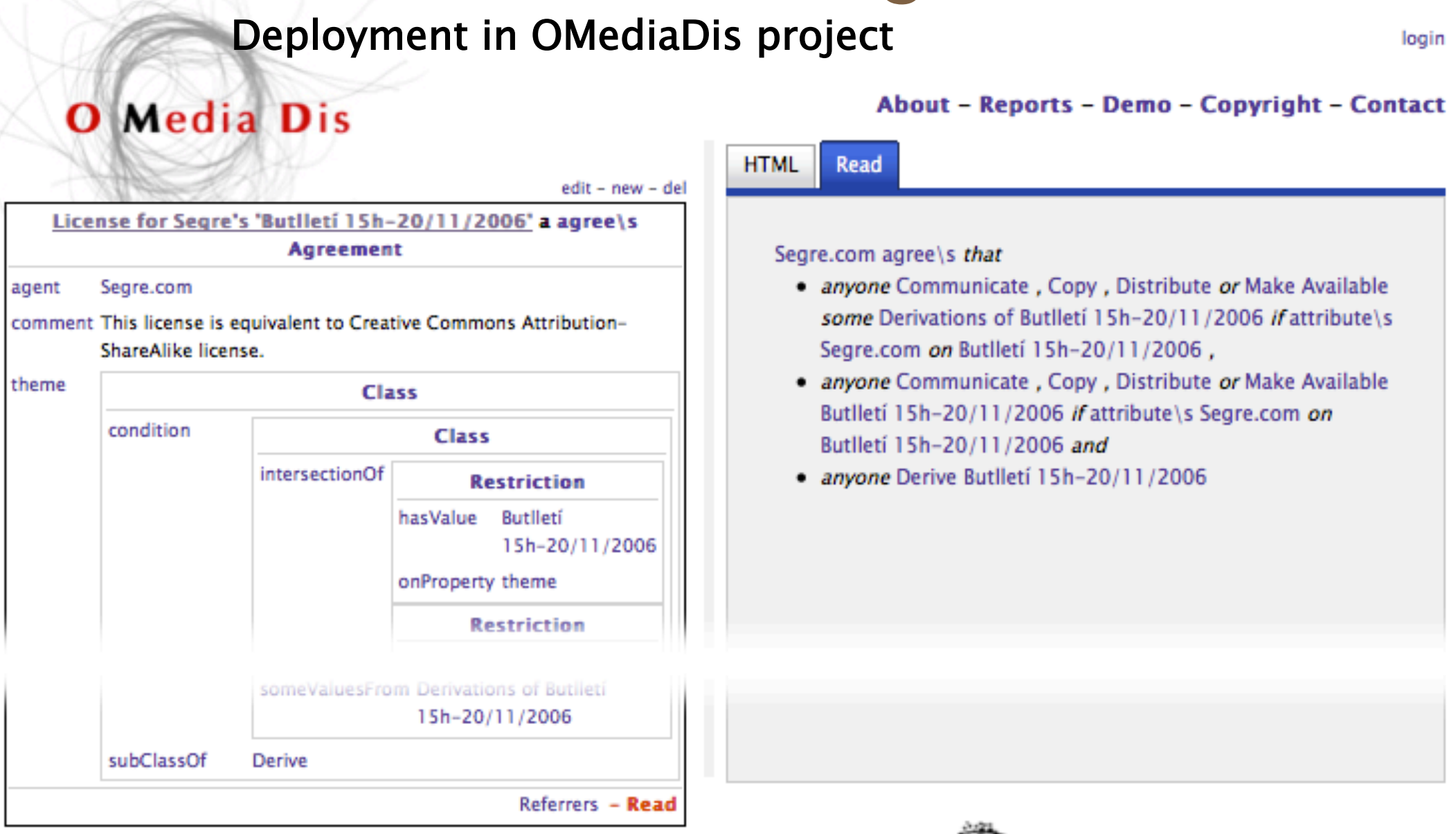
Controlled Natural Language

```
[a co:Agree;
    co:agent :John; co:pointInTime "2007-11-20";
    co:theme [a co:Copy;
                co:agent :Arthur;
                co:theme :The Song;
                co:origin :peerA;
                co:destination:peerB,:peerC;
                co:start "2010-01-01"; co:completion "2010-12-31";
                co:condition [a co:Transfer;
                                co:agent :Peter;
                                co:recipient :John;
                                co:theme :Amount3Euros ]]].
```

"John agree*s*
at 2007-11-20
that Arthur cop*ies* The Song
from peerA **to** peerB *or* peerC
from 2010-01-01 **until** 2010-12-31
with condition Peter transfer*s* amount 3 Euro
to John".

License Rendering

Deployment in OMediaDis project



Conclusions

- CNL + NL-oriented ontology engineering
  - Very expressive REL (ontology)…
  - while quite usable end-user rendering
- Preliminary user test
  - Users explicitly show they preference for CNL rendering by clicking the “Read” service link

Future Work

- Fine tune the case roles to CNL conversion table
- Deal with plurals and verbs (3rd person singular)
- Perform more detailed user tests

Thank you for your attention

rgarcia@diei.udl.cat
http://rhizomik.net/~roberto

HCI & Data Integration
Research Group
Universitat de Lleida