An Institutional Perspective of Semantic Web Stack

Dorel Lucanu

Joint work with Yuan Fang Li and Jin Song Dong from NUS

{dlucanu}@info.uaic.ro

Universitatea “Alexandru Ioan Cuza”, Iaşi


Romania
Iaşi
SLOVAKIA
UKRAINE
HUNGARY
MOLDOVA
ROMANIA
SERBIA AND MONTENEGRO
BULGARIA
Satu Mare
Botoşani
Iasi
Chisinau
Oradea
Cluj-Napoca
Piatra Neamţ
Tîrgu Mureş
Bacău
Arad
Timisoara
Sibiu
Braşov
Galaţi
Brăila
Buzău
Tîrgu Jiu
Ploieşti
Belgrade
(Beograd)
Drobeta-Turnu
Severin
Craiova
Bucharest
(Bucureşti)
Constanţa
47°
45°
23°
27°
0 40 80 mi
0 40 80 km
© maps.com

Outline

- institutions
- institution independent logic programming
- applications to SW
  - logic programming views on web ontologies
  - institutional meaning of RDF layer
  - institutional meaning of layering
  - conclusion

The jungle of Semantic Web Languages

SWRL FOL

Datalog

SWRL

OWL Full−

OWL

F−Logic

OWL Light−

OWL DL

OWL Light

RDF Schema

RDF

OWL Full

DHL

OWL DL−

Description Logic (DL)

DLP

WRL

DAML−OIL

OWL Flight

Motivation

- an integrating mathematical structure for Semantic Web Languages (SWL)
- translating Web ontologies into other formalisms
- a safe way to walk in the jungle
- disputes on layering of SWL
- Open World Assumption (OWA) vs Closed World Assumption (CWA)
- soundness of the reasoners for Web ontologies
- finding the real meaning of Semantic Web Stack

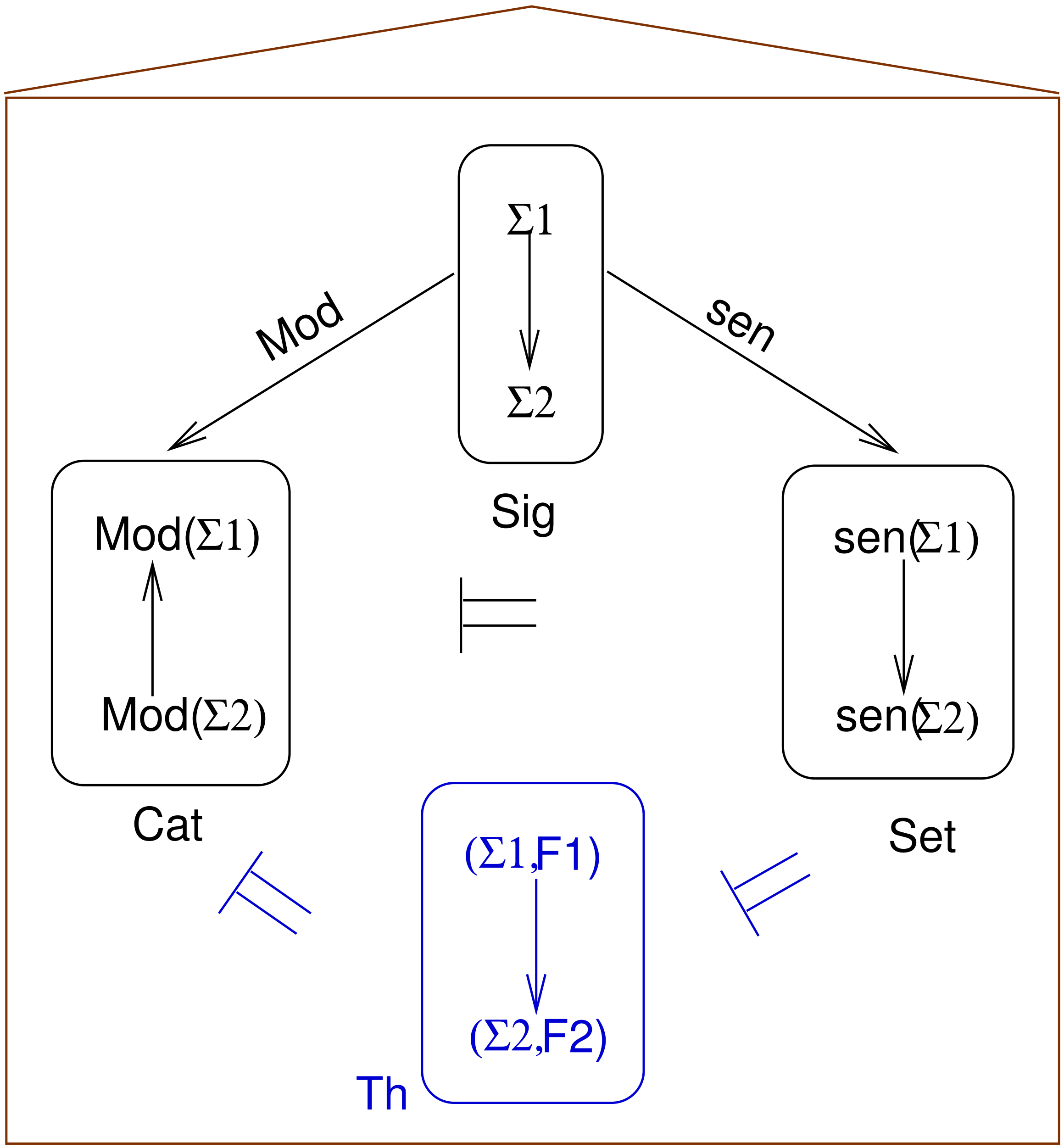
Institutions

- formalize the notion of "a logic"
- study the properties of a logic
  - representation
  - implementation
  - translation of logics

Institutions: ingredients

- **signatures**: formalize vocabularies
- **models**: structures interpreting the symbols (names) from a signature
- **sentences**: formulas built with symbols from signature expressing specific properties
- **satisfaction relation**: says when a given sentence holds in a given model (both correspond to the same signature)

The architecture of an institution



Institutions: signatures

- Horn Logic: $\Sigma = (\mathit{PtN}, \mathit{FtN}, \mathit{CtN})$
  PtN = predicate names, *FtN* = function names, *CtN* = constant names

- Description Logic: $\Sigma = (\mathit{CN}, \mathit{PN}, \mathit{IN})$
  CN = class names, *PN* = property names, *IN* = individual names
  PtN, *FtN*, *CtN* are pairwise disjoint

- OWL: $\Sigma = (\mathit{CN}, \mathit{PN}, \mathit{IN})$
  PtN, *FtN*, *CtN* are pairwise disjoint only for OWL DL and its dialects

Institutions: models

- Horn Logic: Σ-model $\mathbf{I} = (\mathcal{D}_{\mathbf{I}}, \_{\mathbf{I}})$
  $p_{\mathbf{I}} \subseteq \mathcal{D}_{\mathbf{I}}^{arity(p)}$, $f_{\mathbf{I}} : \mathcal{D}_{\mathbf{I}}^{arity(f)} \rightarrow \mathcal{D}_{\mathbf{I}}$, $c_{\mathbf{I}} \in \mathcal{D}_{\mathbf{I}}$

- Description Logic: Σ-model $\mathbf{I} = (\Delta_{\mathbf{I}}, [\![\_]\!]_{\mathbf{I}})$
  $\Delta_{\mathbf{I}}$ domain of the interpretation
  $[\![cn]\!]_{\mathbf{I}} \subseteq \Delta_{\mathbf{I}}$, $[\![pn]\!]_{\mathbf{I}} \subseteq \Delta_{\mathbf{I}} \times \Delta_{\mathbf{I}}$, $[\![in]\!]_{\mathbf{I}} \in \Delta_{\mathbf{I}}$

- OWL: Σ-model $\mathbb{I} = (R_{\mathbb{I}}, S_{\mathbb{I}}, \mathrm{ext}_{\mathbb{I}})$
  $S_{\mathbb{I}} : CN \cup PN \cup IN \rightarrow R_{\mathbb{I}}$
  $\mathrm{ext}_{\mathbb{I}}(cn) \subseteq R_{\mathbb{I}}$, $\mathrm{ext}_{\mathbb{I}}(pn) \subseteq R_{\mathbb{I}} \times R_{\mathbb{I}}$

Institutions: sentences

- Horn Logic: Horn rules $p_1(u_1), \ldots, p_n(u_n) \longrightarrow p_0(u_0)$
  $hasAuthor(p, a) \wedge citedBy(p, q) \longrightarrow CitedAuthor(a)$
- Description Logic:

$$\mathcal{C} ::= \bot \mid \top \mid cn \mid \mathcal{C} \sqcap \mathcal{C} \mid \mathcal{C} \sqcup \mathcal{C} \mid \neg\, \mathcal{C}$$
$$\mid \forall\, pn.\mathcal{C} \mid \exists\, pn.\mathcal{C} \mid \leqslant n\, pn \mid \geqslant n\, pn$$
$$\mathcal{F} ::= \mathcal{C} \sqsubseteq \mathcal{C} \mid \mathcal{C} \equiv \mathcal{C}$$
$$\mid pn^+ \sqsubseteq pn \mid pn \sqsubseteq pn' \mid pn \equiv pn'$$
$$\mid o : \mathcal{C} \mid (o, o') : pn$$

Author $\sqsubseteq$ Person
$\mathtt{Book} \sqsubseteq (\geqslant 1\ \mathtt{hasAuthor})$ (each book has at least one author)

Institutions: sentences

- OWL:
  each book has at least one author

```
<owl:Class rdf:ID="Author">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource=
                      "#hasAuthor" />
      <owl:minCardinality rdf:datatype=
           "#&xsd;nonNegativeInteger">1
      </owl:minCardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

Class(Author partial restriction(hasAuthor minCardinality(1)))

Institutions: satisfaction relation

- relates the models and the sentences: $M \models_\Sigma \varphi$ where M is Σ-model and φ is a Σ-sentence
- it is the subject of the **satisfaction condition** which expresses the invariance of truth under change of notation

$$M' \models_{\Sigma'} \phi(\varphi) \text{ iff } M'\restriction_\phi \models_\Sigma \varphi$$

where $\phi : \Sigma \to \Sigma'$, M' is a Σ'-model, and φ is a Σ-sentence

- DL:

$$\mathsf{I} \models_\Sigma A \sqsubseteq \forall P.B \text{ iff } [\![A]\!]_\mathsf{I} \subseteq \{x \mid (\forall y)(x, y) \in [\![P]\!]_\mathsf{I} \Rightarrow y \in [\![B]\!]_\mathsf{I}\}$$

Institutions: Specifications and Theories

- a **specification** is a pair (Σ, F), where Σ is a signature and F is a set of sentences
- semantical consequences: $(\Sigma, \mathsf{F}) \models \varphi$ iff $(\forall M)(M \models_\Sigma \mathsf{F} \Rightarrow M \models_\Sigma \varphi)$
- a **theory** is a specfication (Σ, F) s.t. $(\forall \varphi)((\Sigma, \mathsf{F}) \models \varphi \Rightarrow \varphi \in \mathsf{F}$
- the inclusion $\mathsf{Th} \rightarrow \mathsf{Spec}$ is an equivalence of categories
- theoroidal (spec-oidal) institutions:
  - signatures are theories (specifications)
  - a (Σ, F)-sentence is a Σ-sentence
  - (Σ, F)-models are Σ-models satisfying F
  - $M \models_{(\Sigma,\mathsf{F})} \varphi$ iff $M \models_\Sigma \varphi$

Relating Institutions

- **morphism**: capture the way in which a “richer” institution is built over a “simpler” one
- **comorphism**: capture the way in which a “simpler” institution is embedded (encoded) into a “richer” one
- both are the subject of a corresponding satisfaction condition
- there exist a variety of definitions for morphisms and variety of definitions for comorphisms in literature
- a prover from the target logic can be used to prove properties from the source logic only if certain conditions are fulfilled

Closed World Assumption

If we cannot prove $(\Sigma, \mathsf{F}) \models \varphi$*, then we add* $\neg\varphi$ *to* F *or, equivalently, restrict the class of* (Σ, F)*-models to those satisfying* $\neg\varphi$*.*

The negation can be defined in an institution independent way:

$$M \models_\Sigma \neg\varphi \text{ iff } M \not\models_\Sigma \varphi$$

$\Im$ *satisfies the closed world assumption (CWA)* iff for each specification $(\Sigma, \mathsf{F}) \in \mathsf{Spec}$,

$$(\Sigma, \mathsf{F}) \models \neg\varphi \text{ iff } (\Sigma, \mathsf{F}) \not\models \varphi$$

Institution independent quantifiers

variables as signature morphisms
$(\forall x)p(x)$

Institution independent quantifiers

variables as signature morphisms

$(\forall\, x)p(x)$

$x : \Sigma = (PtN, FtN, CtN) \rightarrow \Sigma' = (PtN, FtN, CtN \cup \{x\})$

$\mathbf{I} \models_{\Sigma} (\forall\, x)p(x)$ iff

$(\forall\, \mathbf{I}'\ \Sigma'\text{model})\mathbf{I}'{\restriction}_x = \mathbf{I} \Rightarrow \mathbf{I}' \models_{\Sigma'} p(x)$

Institution independent quantifiers

variables as signature morphisms

$(\forall x)p(x)$

$x : \Sigma = (PtN, FtN, CtN) \rightarrow \Sigma' = (PtN, FtN, CtN \cup \{x\})$

$\mathbf{I} \models_{\Sigma} (\forall x)p(x)$ iff

$(\forall \mathbf{I}'\ \Sigma'\text{model})\mathbf{I}'\upharpoonright_x = \mathbf{I} \Rightarrow \mathbf{I}' \models_{\Sigma'} p(x)$

$X : \Sigma \rightarrow \Sigma'$

Institution independent quantifiers

variables as signature morphisms

$(\forall x)p(x)$

$x : \Sigma = (PtN, FtN, CtN) \to \Sigma' = (PtN, FtN, CtN \cup \{x\})$

$\mathbf{I} \models_\Sigma (\forall x)p(x)$ iff

$(\forall \mathbf{I}'\ \Sigma'\text{model})\mathbf{I}'\restriction_x = \mathbf{I} \Rightarrow \mathbf{I}' \models_{\Sigma'} p(x)$

$X : \Sigma \to \Sigma'$

A Σ-sentence $(\forall X)\varphi'$ is the *universal quantification* of the Σ'-sentence φ' iff

$$M \models_\Sigma (\forall X)\varphi' \text{ iff } (\forall M' \text{ a } \Sigma'\text{-model})M'\restriction_X = M \Rightarrow M' \models_{\Sigma'} \varphi'$$

Institution independent Horn Logic

$X : \Sigma \to \Sigma'$ is *representable* iff there is a Σ-model M_X s.t.

1. any Σ-model M' is an expansion along X of a Σ-model M, i. e. $M = M'\upharpoonright_X$, and
2. there is a morphism $M_X \to M$

A set F of Σ-sentences is *basic* iff there is a Σ-model M_{F} s.t. $M \models_\Sigma \mathsf{F}$ iff there is a homomorphism $M_{\mathsf{F}} \to M$.
If $M_{\mathsf{F}} \to M$ is unique, then F is *epic basic*.

A *Horn clause*: $(\forall X)\mathsf{F} \to \mathsf{F}'$ s.t. F is epic basic, F' is basic, and $X : \Sigma \to \Sigma'$ is representable. (Diaconescu, 2004)

Institution independent Horn Logic

DL

- $cn(in)$ is epic basic (cn class name, in individual name)
- $cn_1 \sqcup cn_2(in)$ is not basic
  $\Delta_{\mathbf{I}_1} = \{a\}$, $[\![in]\!]_{\mathbf{I}_1} = a$, $[\![cn_1]\!]_{\mathbf{I}_1} = \{a\}$
  $\Delta_{\mathbf{I}_2} = \{b\}$, $[\![in]\!]_{\mathbf{I}_1} = b$, $[\![cn_2]\!]_{\mathbf{I}_2} = \{b\}$
  $\mathbf{I}_1$ and $\mathbf{I}_2$ cannot be related via a homomorphims

- Lloyd-Topor transformations define a comorphism

  $$(\Phi, \beta, \alpha) : \mathsf{ExtHL} \to \mathsf{HL}^{\wedge}$$

  $\mathsf{HL}^{\wedge}$ is HL enriched s.t. any conjunction of rules is a sentence

Institution independent Logic Programming

OWA Logic Programming institution with base institution $\Im$, $\mathcal{LP}(\Im)$, is defined as follows:

1. the signatures are Horn specifications HSpec;
2. the model functor is $\mathsf{Mod}(\Im)$ extended to Horn specifications;
3. the Σ-sentences are *Horn queries* $(\exists X)\varphi$ with φ basic;
4. the satisfaction is given by $M \models_{(\Sigma,\mathsf{F})} (\exists X)\varphi$ iff $M \models_{\Sigma} (\exists X)\varphi$, i.e., the satisfaction is inherited from the base institution.

Institution independent Logic Programming

CWA Logic Programming institution with base institution $\Im$, $\mathcal{LP}^{\triangle}(\Im)$, is defined similarly to $\mathcal{LP}(\Im)$ except the model functor which associates a category of logically equivalent canonical models $\triangle(\Sigma, \mathsf{F})$ with each Horn specification (Σ, F).

Herbrand Theorem (Diaconescu, 2004). In an arbitrary institution consider a specification (Σ, F) which has an initial model $0_{\Sigma,\mathsf{F}}$. Then for each query $(\exists X)\varphi$

$$(\Sigma, \mathsf{F}) \models (\exists X)\varphi \text{ iff } 0_{\Sigma,\mathsf{F}} \models (\exists X)\varphi$$

Institution independent Logic Programming

The institution of a Logic Programming with Knowledge Bases, $\mathcal{LP}^{kb}(\Im)$, is defined as follows:

1. a signature is a pair $((\Sigma, \mathsf{F}), \mathit{KB})$, where (Σ, F) is a Horn specification (the logic program), and *KB* is a (Σ, F)-knowledge base (Grothendieck construction);
2. the model functor maps each signature $((\Sigma, \mathsf{F}), \mathit{KB})$ into a category $\triangle(\Sigma_{KB}, \mathsf{F} \cup \mathsf{F}_{KB})$;
3. the sentences are *Horn queries*;
4. the satisfaction is given by $\triangle(\Sigma, \mathsf{F}, \mathit{KB}) \models_{(\Sigma,\mathsf{F})} (\exists X)\varphi$ iff $\triangle(\Sigma, \mathsf{F}, \mathit{KB}) \models_{\Sigma} (\exists X)\varphi$.

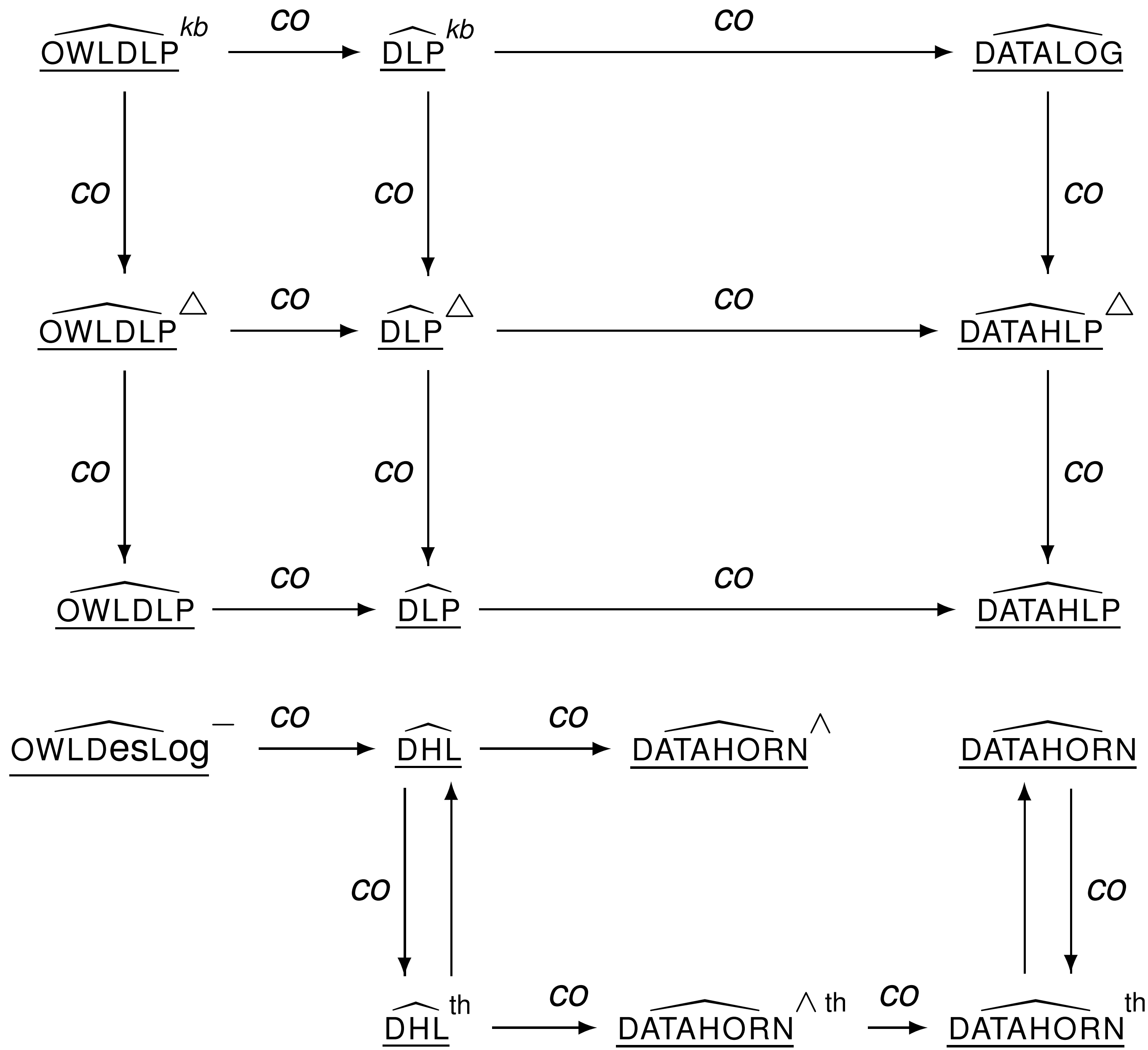
Logic Programming Views

- $\underline{\widehat{\mathrm{OWLDLP}}}$ is OWA LP over $\underline{\widehat{\mathrm{OWLDesLog}}}^{-}$
- $\underline{\widehat{\mathrm{OWLDLP}}}^{\triangle}$ is CWA LP over $\underline{\widehat{\mathrm{OWLDesLog}}}^{-}$
- $\underline{\widehat{\mathrm{OWLDLP}}}^{kb}$ is LP with KB over $\underline{\widehat{\mathrm{OWLDesLog}}}^{-}$

- $\underline{\widehat{\mathrm{DLP}}}$ is OWA LP over $\underline{\widehat{\mathrm{DHL}}}$
- $\underline{\widehat{\mathrm{DLP}}}^{\triangle}$ is CWA LP over $\underline{\widehat{\mathrm{DHL}}}$
- $\underline{\widehat{\mathrm{DLP}}}^{kb}$ is LP with KB over $\underline{\widehat{\mathrm{DHL}}}$

- $\underline{\widehat{\mathrm{DATAHLP}}}$ is OWA LP over $\underline{\widehat{\mathrm{DATAHORN}}}$
- $\underline{\widehat{\mathrm{DATAHLP}}}^{\triangle}$ is CWA LP over $\underline{\widehat{\mathrm{DATAHORN}}}$
- $\underline{\widehat{\mathrm{DATALOG}}}$ is LP with KB over $\underline{\widehat{\mathrm{DATAHORN}}}$

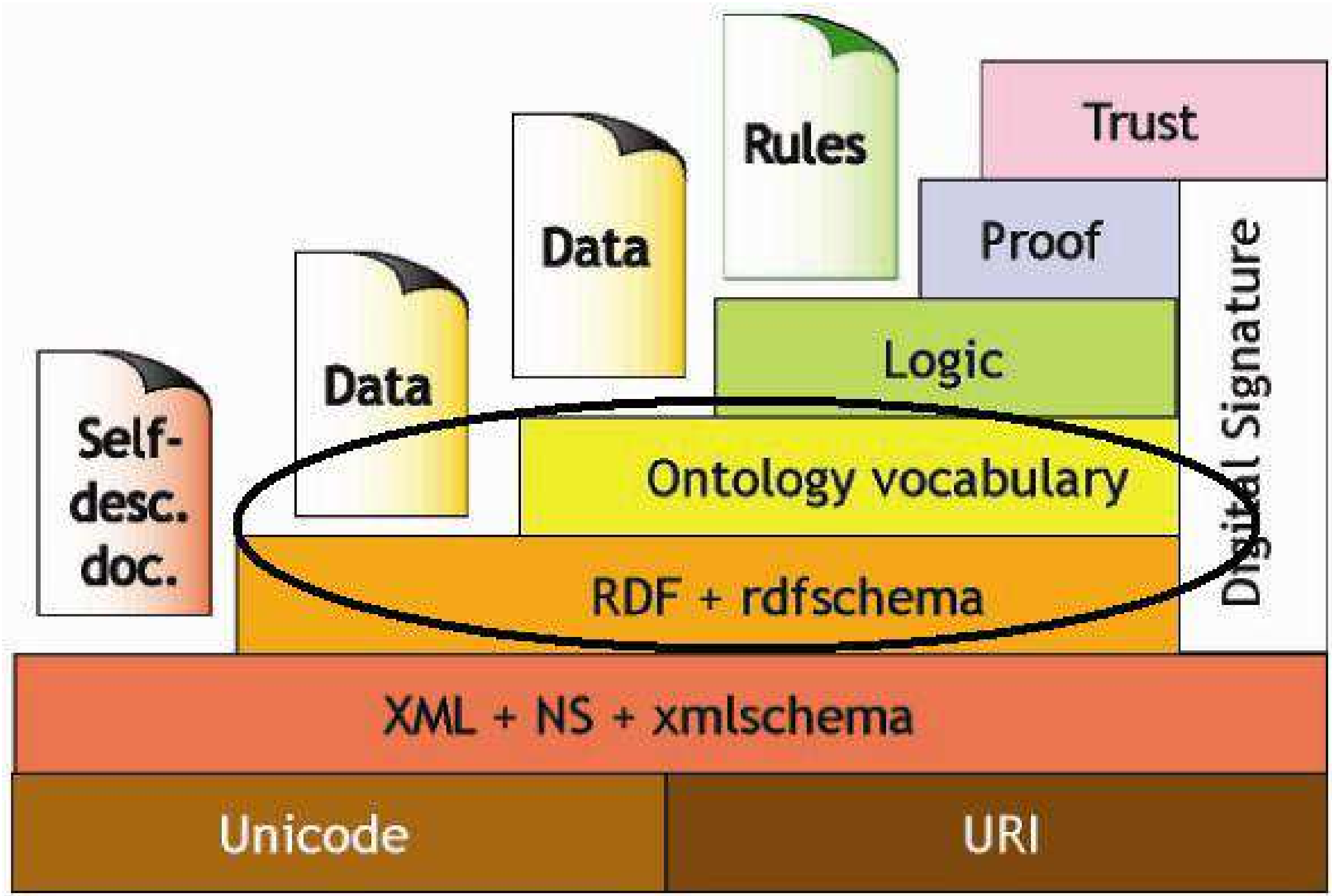
Logic Programming Views on OWL DL$^-$



Adding rules to Web Ontology Languages . . .

$$
\begin{array}{ccccc}
\mathcal{LP}^{kb}(\widehat{\underline{\mathsf{OWLDesLog}}}^{-}) & \xrightarrow{co} & \mathcal{LP}^{kb}(\widehat{\underline{\mathsf{DHL}}}) & \xrightarrow{co} & \widehat{\underline{\mathsf{DATAHLP}}}^{kb} \\
\downarrow co & & \downarrow co & & \downarrow co \\
\mathcal{LP}^{\triangle}(\widehat{\underline{\mathsf{OWLDesLog}}}^{-}) & \xrightarrow{co} & \mathcal{LP}^{\triangle}(\widehat{\underline{\mathsf{DHL}}}) & \xrightarrow{co} & \widehat{\underline{\mathsf{DATAHLP}}}^{\triangle} \\
\downarrow co & & \downarrow co & & \downarrow co \\
\mathcal{LP}(\widehat{\underline{\mathsf{OWLDesLog}}}^{-}) & \xrightarrow{co} & \mathcal{LP}(\widehat{\underline{\mathsf{DHL}}}) & \xrightarrow{co} & \widehat{\underline{\mathsf{DATAHLP}}} \\
 & & & & \\
\mathcal{HL}(\widehat{\underline{\mathsf{OWLDesLog}}}^{-}) & \longrightarrow & \mathcal{HL}(\widehat{\underline{\mathsf{DHL}}}) & \longrightarrow & \widehat{\underline{\mathsf{DATAHORN}}}^{\wedge}
\end{array}
$$

RDF Serialization



From Semantic Web talk by Tim Berners-Lee at XML 2000

Bare RDF logic $\widehat{\underline{\mathsf{BRDF}}}$

- signatures: RR a set of resources references
- models: $\mathbb{I} = (R_{\mathbb{I}}, P_{\mathbb{I}}, S_{\mathbb{I}}, \mathrm{ext}_{\mathbb{I}})$, where
  $R_{\mathbb{I}}$ is a set of resources,
  $P_{\mathbb{I}} \subseteq R_{\mathbb{I}}$ - the set of properties,
  $S_{\mathbb{I}} : RR \to R_{\mathbb{I}}$
  $\mathrm{ext}_{\mathbb{I}} : P_{\mathbb{I}} \to \mathcal{P}(R_{\mathbb{I}} \times R_{\mathbb{I}})$ is an extension function mapping each property to a set of pairs of resources that it relates
- RR-sentences are triples of the form (sn, pn, on), where $sn, pn, on \in RR$
- satisfaction is defined as follows:

$$\mathbb{I} \models_{RR} (sn, pn, on) \; iff \, (S_{\mathbb{I}}(sn), S_{\mathbb{I}}(on)) \in \mathrm{ext}_{\mathbb{I}}(\mathrm{ext}_{\mathbb{I}}(pn)),$$

RDF theory $\mathtt{RDF} = (\mathtt{RDFVoc}, T_{\mathtt{RDF}})$

`RDFVoc` includes the following items:

```
rdf:type, rdf:Property, rdf:value,
rdf:Statement, rdf:subject, rdf:predicate, rdf:object,
rdf:List, rdf:first, rdf:rest, rdf:nil,
rdf:Seq, rdf:Bag, rdf:Alt, rdf:li rdf:_1 rdf:_2 ...
```

and $T_{\mathtt{RDF}}$ is

```
{
  (rdf:type, rdf:type, rdf:Property),
  (rdf:subject, rdf:type, rdf:Property),
  (rdf:predicate, rdf:type, rdf:Property),
  (rdf:object, rdf:type, rdf:Property),
  (rdf:value, rdf:type, rdf:Property),
  (rdf:first, rdf:type, rdf:Property),
  (rdf:rest rdf:type rdf:Property),
  (rdf:nil rdf:type rdf:List),
  (rdf:_1 rdf:type rdf:Property),
  (rdf:_2 rdf:type rdf:Property),
  ...
}
```

RDF theory $\mathtt{RDF} = (\mathtt{RDFVoc}, T_{\mathtt{RDF}})$

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  ...
```

The prefix `rdf:` is used for the namespace

`URI: http://www.w3.org/1999/02/22-rdf-syntax-ns#`

We suppose that there is given a set $R_{\mathbb{RDF}}$ of RDF resources and a function $S_{\mathbb{RDF}} : \mathtt{RDFVoc} \to R_{\mathbb{RDF}}$ which associates a resource with each RDF symbol. It is easy to see that $R_{\mathbb{RDF}}$ and $S_{\mathbb{RDF}}$ can be extended to a $\mathtt{RDF}$-model $\mathbb{RDF}$.

RDF logic $\widehat{\underline{\mathtt{RDF}}}$

- signatures: theory morphisms $f : \mathtt{RDF} \rightarrow (RR, T)$
- models: (RR, T)-models $\mathbb{I}$ such that
  - $R_{\mathbb{I}}$ includes $R_{\mathbb{RDF}}$ and the restriction of $S_{\mathbb{I}}$ to $\mathtt{RDFVoc}$ coincides with $S_{\mathbb{RDF}}$,
  - if $p \in P_{\mathbb{I}}$ then $(p, \mathtt{rdf} : \mathtt{Property}) \in \mathrm{ext}_{\mathbb{I}}(\mathtt{rdf} : \mathtt{type})$
- f-sentences are RR-sentences
- satisfaction: $\mathbb{I} \models_f (sn, pn, on)$ iff $\mathbb{I} \models_{RR} (sn, pn, on)$

RDF Schema theory $\mathtt{RDFS} = (\mathtt{RDFSVoc}, T_{\mathtt{RDFSVoc}})$

RDFSVoc is RDFVoc together with

```
rdfs:domain, rdfs:range, rdfs:Resource,
rdfs:Literal, rdfs:Datatype,
rdfs:Class, rdfs:subClassOf,
rdfs:subPropertyOf,
rdfs:member, rdfs:Container, rdfs:ContainerMembershipProperty
```

and the sentences $T_{\mathtt{RDFS}}$

```
{
  rdf:type, rdfs:domain, rdfs:Resource,
  rdfs:domain, rdfs:domain, rdf:Property,
  rdfs:range, rdfs:domain, rdf:Property,
  rdfs:subPropertyOf, rdfs:domain, rdf:Property,
  rdfs:subClassOf, rdfs:domain, rdfs:Class,
  rdf:subject, rdfs:domain, rdf:Statement,
  ...
}
```

RDF Schema theory $\mathtt{RDFS} = (\mathtt{RDFSVoc}, T_{\mathtt{RDFSVoc}})$

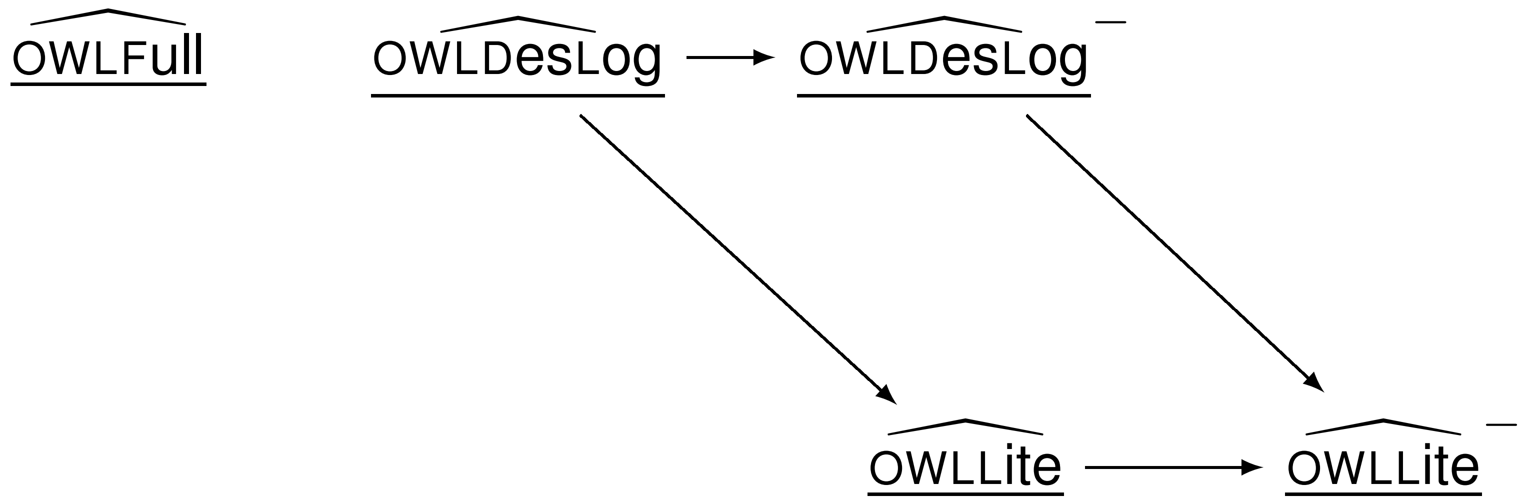
```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  ...
```

We suppose that there is given a set $R_{\mathbb{RDFS}}$ of RDF Schema resources and a function $S_{\mathbb{RDFS}} : \mathtt{RDFSVoc} \to R_{\mathbb{RDFS}}$ which associates a resource with each RDF Schema symbol and that satisfies $S_{\mathbb{RDFS}}|_{\mathtt{RDFVoc}} = S_{\mathbb{RDF}}$.

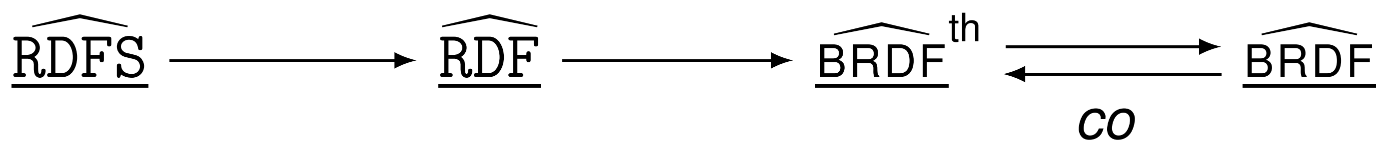
RDF Schema logic $\widehat{\mathtt{RDFS}}$

- signatures: theory morphisms $f : \mathtt{RDFS} \longrightarrow (RR, T)$
- models: (RR, T)-models $\mathbb{I}$ such that
  - $R_{\mathbb{I}}$ includes $R_{\mathbb{RDFS}}$ and the restriction of $S_{\mathbb{I}}$ to $\mathtt{RDFSVoc}$ coincides with $S_{\mathbb{RDFS}}$
  - $\mathrm{ext}_{\mathbb{I}}(\mathtt{rdfs:Resource}) = R_{\mathbb{I}}$
  - $(\forall x, y, u, v \in R_{\mathbb{I}})(x, y) \in \mathrm{ext}_{\mathbb{I}}(\mathtt{rdfs:domain}) \wedge (u, v) \in \mathrm{ext}_{\mathbb{I}}(x) \Rightarrow u \in \mathrm{ext}_{\mathbb{I}}(y)$
  - $(\forall x, y, u, v \in R_{\mathbb{I}})(x, y) \in \mathrm{ext}_{\mathbb{I}}(\mathtt{rdfs:range}) \wedge (u, v) \in \mathrm{ext}_{\mathbb{I}}(x) \Rightarrow v \in \mathrm{ext}_{\mathbb{I}}(y)$
  - $(\forall x, y \in R_{\mathbb{I}})(x, y) \in \mathrm{ext}_{\mathbb{I}}(\mathtt{rdfs:subClassOf}) \Rightarrow \mathrm{ext}_{\mathbb{I}}(x) \subseteq \mathrm{ext}_{\mathbb{I}}(y)$
  - $(\forall x \in \mathrm{ext}_{\mathbb{I}}(\mathtt{rdf:Class}))(x, \mathtt{rdfs:Resource}) \in \mathrm{ext}_{\mathbb{I}}(\mathtt{rdfs:subClassOf}))$
  - $(\forall x, y \in R_{\mathbb{I}})(x, y) \in \mathrm{ext}_{\mathbb{I}}(\mathtt{rdfs:subPropertyOf}) \Rightarrow \mathrm{ext}_{\mathbb{I}}(x) \subseteq \mathrm{ext}_{\mathbb{I}}(y)$
  - $(\forall x \in \mathrm{ext}_{\mathbb{I}}(\mathtt{rdfs:ContainerMembershipProperty}))$ $(x, \mathtt{rdfs:member}) \in \mathrm{ext}_{\mathbb{I}}(\mathtt{rdfs:subPropertyOf})$
- f-sentences are RR-sentences
- satisfaction: $\mathbb{I} \models_f (sn, pn, on)$ iff $\mathbb{I} \models_{RR} (sn, pn, on)$

RDF and Web Ontology Vocabulary layers



Web Ontology Vocabulary layer



RDF layer

OWL DL theory

$\texttt{OWLDL} = (\texttt{OWLDLVoc}, T_{\texttt{OWLDL}})$, where `OWLDLVoc` is `OWLLVoc` together with

```
owl:DepricatedClass, owl:DisjointClasses, owl:SubClassOf,
owl:Functional,  owl:InverseFunctional, owl:Transitive,
owl:SameIndividual, DifferentIndividuals, owl:someValues,
owl:Thing, owl:Nothing,
owl:intersectionOf, owl:unionOf, owl:complementOf, owl:oneOf,
owl:someValues, owl:hasValue, owl:maxCardinality
```

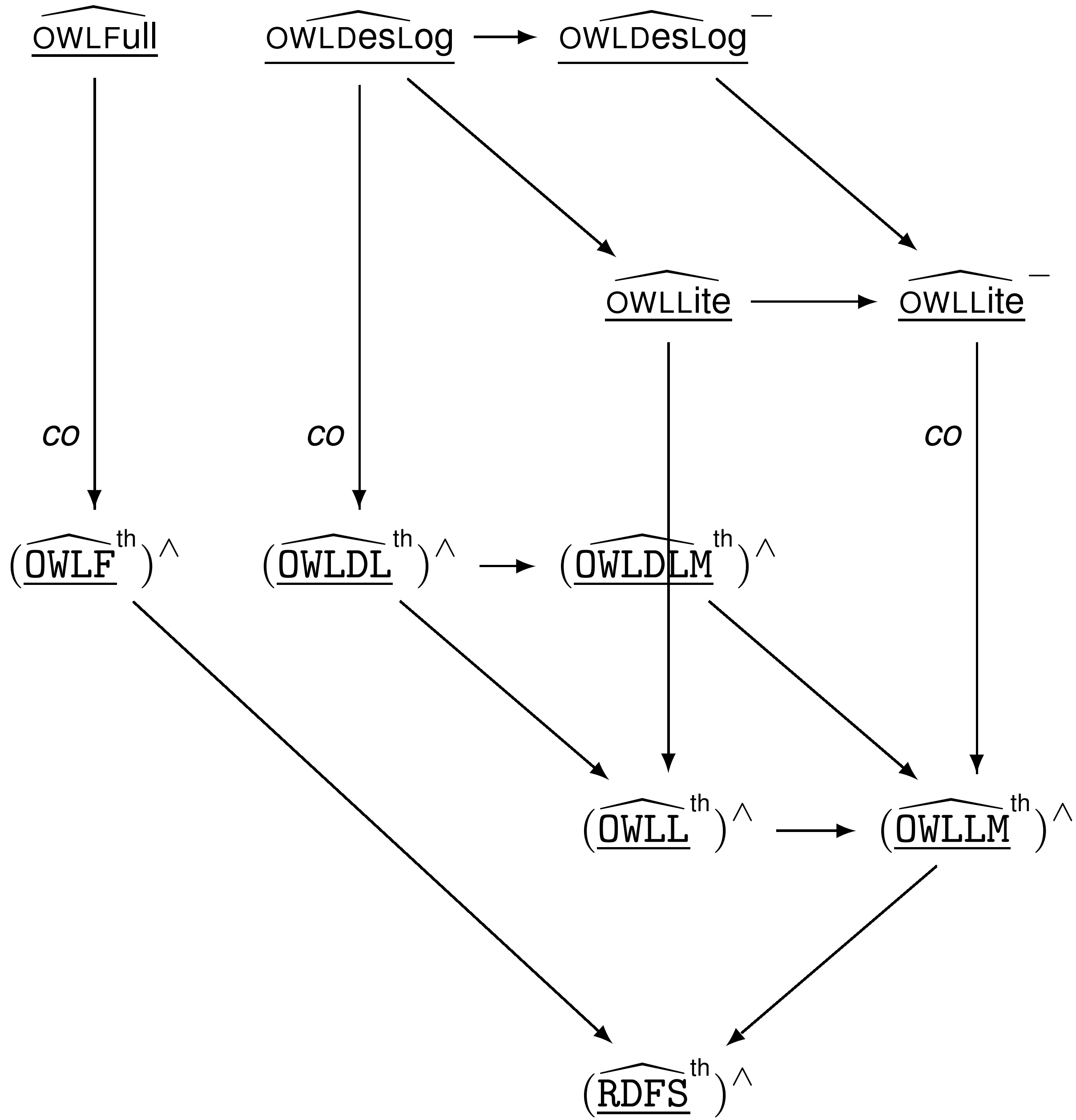
and $T_{\texttt{OWLDL}}$ is

```
{
  (owl:intersectionOf, rdf:type, rdf:Property),
  (owl:intersectionOf, rdfs:domain, owl:Class),
  (owl:intersectionOf, rdfs:range, rdf:List),
  (owl:equivalentClass, rdf:type, rdf:Property),
  (owl:equivalentClass, rdfs:subPropertyOf, rdfs:subClassOf),
  (owl:equivalentClass, rdfs:domain, owl:Class),
  (owl:equivalentClass, rdfs:range, owl:Class),
  (owl:disjointWith, rdf:type, rdf:Property),
  (owl:disjointWith, rdfs:domain, owl:Class),
  (owl:disjointWith, rdfs:range, owl:Class),
  ...
}
```

RDF Serialization of OWL DL: $\widehat{\underline{\texttt{OWLDL}}}$

- signatures: theory morphisms $f : \texttt{OWLDL} \rightarrow (RR, T)$
- models: (RR, T)-models $\mathbb{I}$ such that
  - $R_{\mathbb{I}}$ includes $R_{\mathbb{OWLDL}}$ and the restriction of $S_{\mathbb{I}}$ to $\texttt{OWLDLVoc}$ coincides with $S_{\mathbb{OWLDL}}$.
  - Restrictions expressing the intended meaning of the new features.
- f-sentences are RR-sentences
- satisfaction: $\mathbb{I} \models_f (sn, pn, on)$ iff $\mathbb{I} \models_{RR} (sn, pn, on)$

The meaning of layering



Institutions: Main references

- Introducing Institutions, by J. Goguen and R. Burstall, 1984
- Institutions: Abstract model theory for specification and programming, by J. Goguen and R. Burstall, 1992
- Structuring theories on consequence, by J. Fiadeiro and A.Sernadas - 1988
- May I Borrow Your Logic?, by M. Cerioli and J. Meseguer,1993
- Moving Between Logical Systems, Andrzej Tarlecki, 1995
- Institution Morphisms, by J. Goguen and Gr. Rosu, 2002
- Grothendieck Institutions, by R. Diaconescu, 2002
- Herbrand Theorem, by R. Diaconescu, 2004

Conclusion

- institutions offer
  - a rigurous and systematic approach of the logics underlying SW languages
  - an important step towards structuring and re-using ontology parts
  - a solid framework for relating SW languages with other formalisms and for proving the soundness of the reasoners

Questions?

Thank you!