

The Distributed Ontology, Modeling and Specification Language (DOL)

Till Mossakowski¹ Oliver Kutz¹
Christoph Lange² Mihai Codrescu¹



OTTO VON GUERICKE
UNIVERSITÄT
MAGDEBURG

INF

FAKULTÄT FÜR
INFORMATIK

¹University of Magdeburg

²University of Bonn

IFIP WG 1.3 meeting, 2014-01-09

Motivation

The Big Picture of Interoperability

Modeling	Specification	Knowledge engineering
Objects/data	Software	Concepts/data
Models	Specifications	Ontologies
Metamodels	Specification languages	Ontology languages

Diversity and the need for interoperability occur at all these levels!
(Formal) ontologies, (formal) specifications and (formal) models will henceforth be abbreviated as **OSMs**.

Ontology use Case: OMG's Date-Time Vocabulary

- date-time vocabulary is formulated in different languages: SBVR, Common Logic, IKL, UML+OCL, OWL
- different languages address different audiences
 - SBVR: **business users**
 - UML+OCL: **software implementors**
 - OWL: **ontology developers and users**
 - Common Logic, IKL: (**foundational**) ontology developers and users
- How can we
 - formally relate the different logical specifications?
 - specify the OWL version to be an approximation of the Common Logic version?
 - extract submodules covering specific aspects?

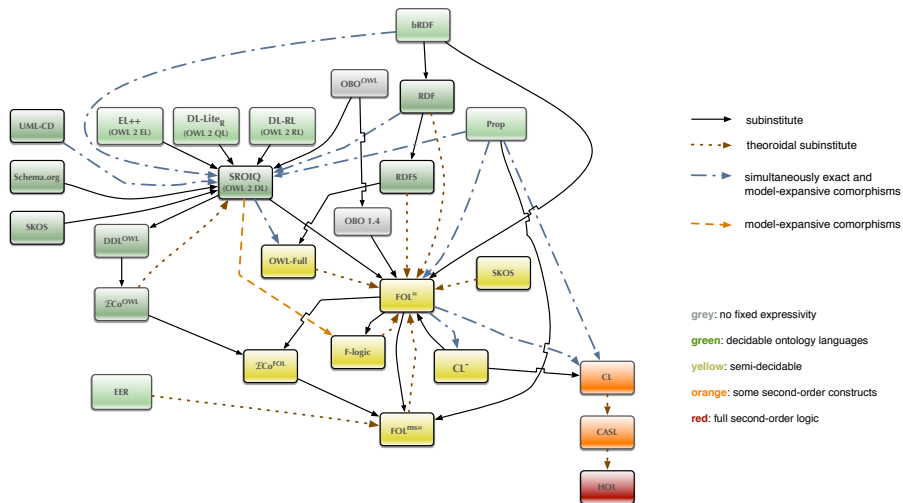
Use Case: Refinement of specifications

- refinement from requirements to design to code
- many different formalisms
- formalism may change during formal development
- yet, some general mechanism of refinements are always the same

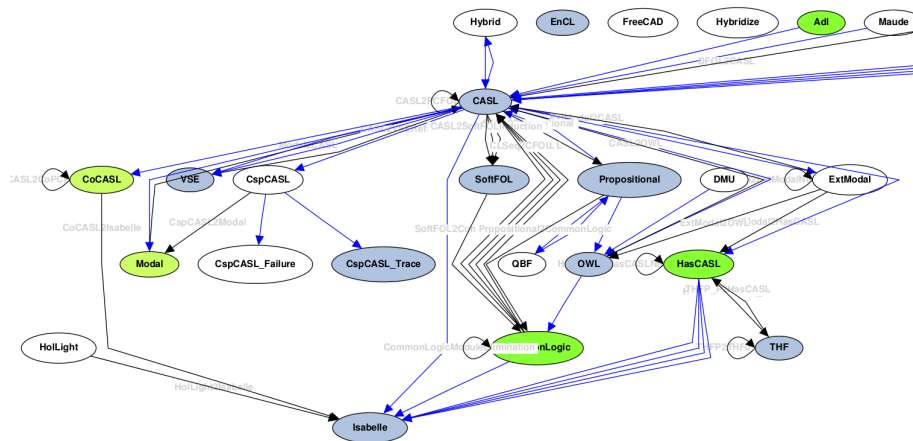
Use Case: Consistency and satisfiability among UML models

- does an object diagram satisfy a class diagram?
- Does a state machine satisfy an OCL specification?
- Do the protocol state machines at the ends of a connector fit together?
- Does a state machine refine the protocol state machines in a structure diagram?

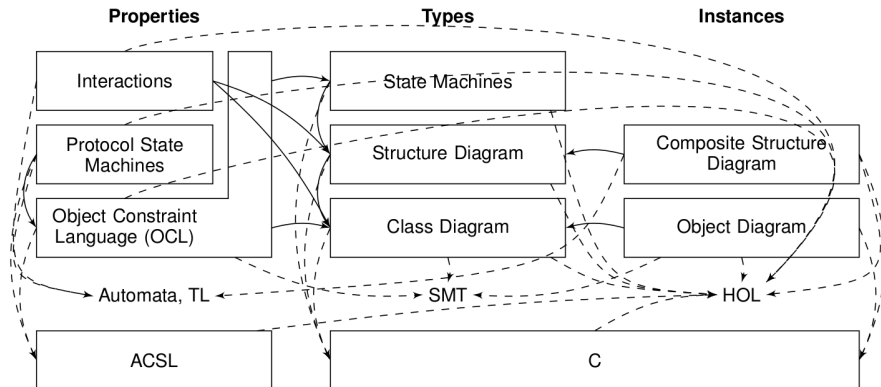
Ontologies: An Initial Logic Graph



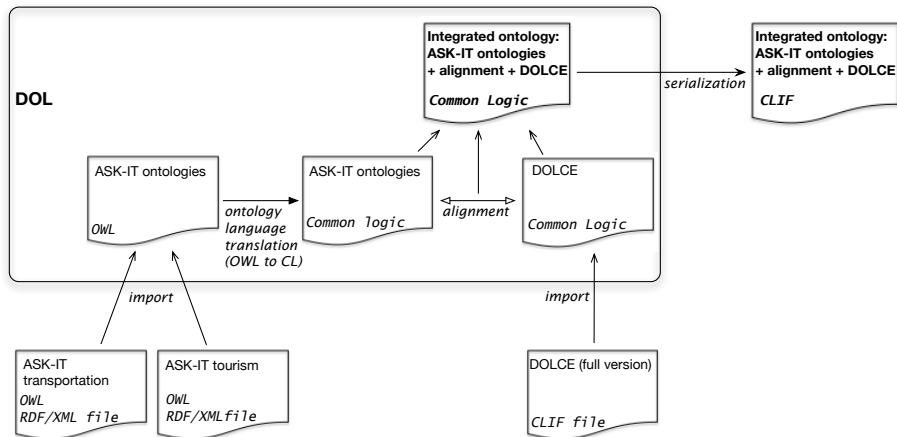
Specifications: An Initial Logic Graph



UML models: An Initial Logic Graph



Sample Use of DOL



Motivation: Diversity of Operations on and Relations among OSMs

Various operations and relations on OSMs are in use:

- structuring: union, translation, hiding, ...
- refinement
- matching and alignment
 - of many OSMs covering one domain
- module extraction
 - get relevant information out of large OSM
- approximation
 - model in an expressive language, reason fast in a lightweight one
- ontology-based database access/data management
- distributed OSMs
 - bridges between different modellings

OntolOp

Need for a Unifying Meta Language

Not yet another ontology language, but a meta language covering

- diversity of ontology languages
- translations between these
- diversity of operations on and relations among OSMs

Current standards like the OWL API or the alignment API only cover parts of this

The
Ontology, Modeling and Specification
Integration and Interoperability (OntoOp)
initiative addresses this

The OntoOp initiative

- started in 2011 as ISO 17347 within ISO/TC 37/SC 3
- now continued as OMG standard
 - OMG has more experience with **formal semantics**
 - OMG documents will be **freely available**
 - focus extended from ontologies only to **formal models** and **specifications** (i.e. logical theories)
 - request for proposals (RFP) to be issued in fall 2013
 - proposals answering RFP due in December 2014
- 50 experts participate, ~ 15 have contributed
- OntoOp is open for your ideas, so **join us!**

Requirements in the OMG RFP OntoOp

- provide a **meta-language** for:
 - **logically heterogeneous** OSMs
 - **modular** OSMs
 - **module extraction**, **approximation**
 - **links** (interpretations, alignments) between OSMs/modules
 - **combination** of OSMs along links
- provide an **abstract syntax** as MOF or SMOF model
- provide a **concrete syntax**
- provide a **formal semantics**
 - criteria for logics to conform with OntoOp
 - translations between these logics
- be **logic-agnostic**, e.g. OSMs consist of symbols and axioms

DOL

The Distributed Ontology, Modeling and Specification Language (DOL)

- has been prepared within ISO/TC 37/SC 3
- now continued as a proposal for the OMG RFP OntoOp
 - DOL = one specific answer to the RFP requirements
 - there may be other answers to the RFP (but unlikely)
- DOL is based on some **graph of institutions and (co)morphisms**
- DOL has a **model-theoretic semantics**

Related work

- Structured specifications and their semantics (Clear, ASL, CASL, ...)
- Heterogeneous specification (HetCASL)
- modular ontologies (WoMo workshop series)

Institutions

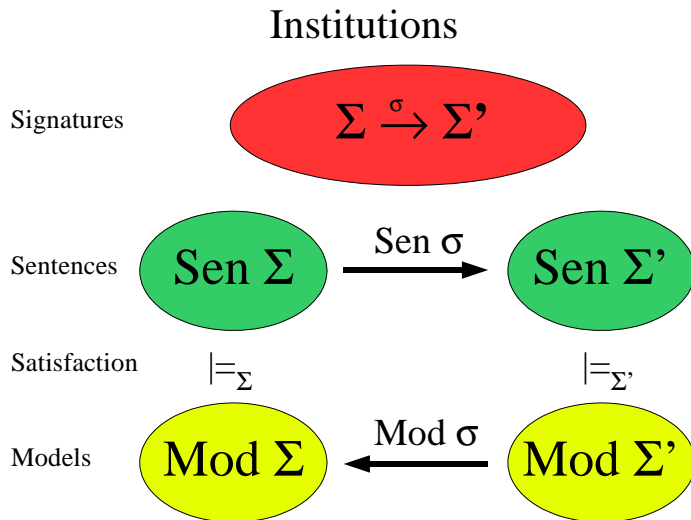
An **institution** \mathcal{I} consists of:

- a category $\mathbf{Sign}_{\mathcal{I}}$ of **signatures**;
- a functor $\mathbf{Sen}_{\mathcal{I}}: \mathbf{Sign}_{\mathcal{I}} \rightarrow \mathbf{Set}$, giving a set $\mathbf{Sen}(\Sigma)$ of **Σ -sentences** for each signature $\Sigma \in |\mathbf{Sign}_{\mathcal{I}}|$, and a function $\mathbf{Sen}(\sigma): \mathbf{Sen}(\Sigma) \rightarrow \mathbf{Sen}(\Sigma')$ that yields **σ -translation** of Σ -sentences to Σ' -sentences for each $\sigma: \Sigma \rightarrow \Sigma'$;
- a functor $\mathbf{Mod}_{\mathcal{I}}: \mathbf{Sign}_{\mathcal{I}}^{op} \rightarrow \mathbf{Set}$, giving a set $\mathbf{Mod}(\Sigma)$ of **Σ -models** for each signature $\Sigma \in |\mathbf{Sign}_{\mathcal{I}}|$, and a functor $-\downarrow_{\sigma} = \mathbf{Mod}(\sigma): \mathbf{Mod}(\Sigma') \rightarrow \mathbf{Mod}(\Sigma)$; for each $\sigma: \Sigma \rightarrow \Sigma'$;
- for each $\Sigma \in |\mathbf{Sign}_{\mathcal{I}}|$, a **satisfaction relation**
 $\models_{\mathcal{I}, \Sigma} \subseteq \mathbf{Mod}_{\mathcal{I}}(\Sigma) \times \mathbf{Sen}_{\mathcal{I}}(\Sigma)$

such that for any signature morphism $\sigma: \Sigma \rightarrow \Sigma'$, Σ -sentence $\varphi \in \mathbf{Sen}_{\mathcal{I}}(\Sigma)$ and Σ' -model $M' \in \mathbf{Mod}_{\mathcal{I}}(\Sigma')$:

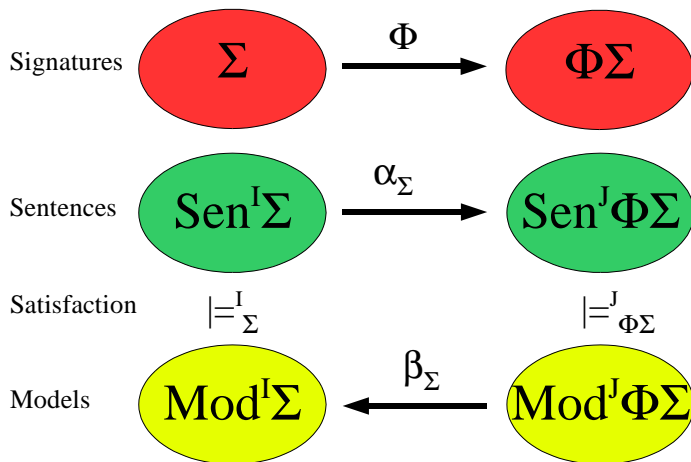
$$M' \models_{\mathcal{I}, \Sigma'} \sigma(\varphi) \iff M'|_{\sigma} \models_{\mathcal{I}, \Sigma} \varphi \quad [\text{Satisfaction condition}]$$

Institutions



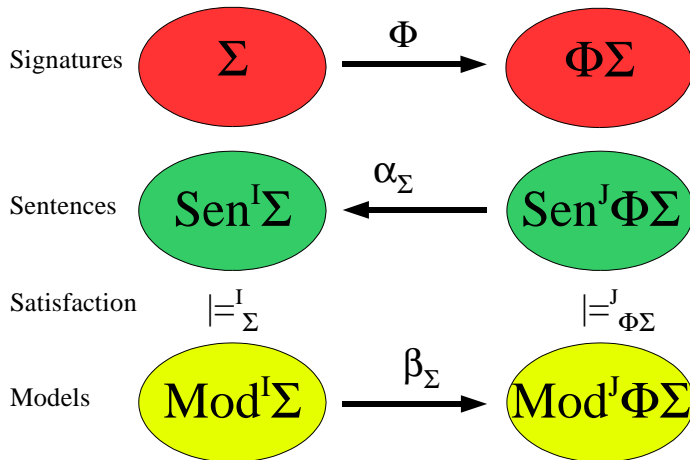
Institution comorphisms (embeddings, encodings)

Institution comorphisms



Institution morphisms (projections)

Institution morphisms



Unions, differences and inclusive institutions

We assume that for each institution, there exists (possibly partial) union and difference operations on signatures. E.g. an inclusion system on signatures would be a good framework where this can be required.

Definition (Gogen, Roşu)

An *inclusive category* is a category having a broad subcategory which is a partially ordered class.

An *inclusive institution* is one with an inclusive signature category such that the sentence functor preserves inclusions.

We also assume that model categories are inclusive.

Overview of DOL

1 modular and heterogeneous OSMs

- basic OSMs (flattenable)
- extensions, unions, translations (flattenable)
- reductions (elusive)
- approximations, module extractions (flattenable)
- minimization, maximization (elusive)
- combination (flattenable)
- OSM bridges (flattenable)

only OSMs with flattenable components are flattenable

2 distributed OSMs (based on 1)

- OSM definitions (giving a name to an OSM)
- interpretations (of theories)
- equivalences
- module relations
- alignments

Semantic domains of DOL

- semantics of a flattenable OSM has form (I, Σ, Ψ) (**theory-level**)
- semantics of an elusive OSM has form (I, Σ, \mathcal{M}) (**model-level**)
 - institution I
 - signature Σ in I
 - set Ψ of Σ -sentences
 - class \mathcal{M} of Σ -models

We can obtain the model-level semantics from the theory-level semantics by taking $\mathcal{M} = \{M \in \mathbf{Mod}(\Sigma) \mid M \models \Psi\}$.

- **semantics of a distributed OSM** has form
 $\Gamma: IRI \longrightarrow (OSM \uplus OSM \times OSM \times SigMor)$
 - OSM is the class of all triples (I, Σ, Ψ) , (I, Σ, \mathcal{M})
 - for interpretations etc., domain, codomain and signature morphism is recorded: $OSM \times OSM \times SigMor$

Modular and Heterogeneous OSMs

Basic OSMs

- written in **some OSM language** from the logic graph
- semantics is **inherited** from the OSM language
- e.g. in OWL:

Class: Woman **EquivalentTo:** Person **and** Female
ObjectProperty: hasParent

- e.g. in Common Logic:

```
(cl-module PreOrder
  (forall (x) (le x x))
  (forall (x y z)
    (if (and (le x y)
              (le y z))
        (le x z))))
```

Semantics of basic OSMs

We assume that $\llbracket O \rrbracket_{basic} = (I, \Sigma, \Psi)$ for some OSM language based on I

- institution I
- signature Σ in I
- set Ψ of Σ -sentences

This direct leads to a theory-level semantics for the OSM.

Extensions

- O_1 **then** O_2 : extension of O_1 by new symbols and axioms O_2
- O_1 **then %mcons** O_2 : model-conservative extension
 - each O_1 -model has an expansion to O_1 **then** O_2
- O_1 **then %ccons** O_2 : consequence-conservative extension
 - O_1 **then** $O_2 \models \varphi$ implies $O_1 \models \varphi$, for φ in the language of O_1
- O_1 **then %def** O_2 : definitional extension
 - each O_1 -model has a **unique** expansion to O_1 **then** O_2
- O_1 **then %implied** O_2 : like %mcons, but O_2 must not extend the signature
- example in OWL:

```

Class Person
Class Female
then %def
Class: Woman EquivalentTo: Person and Female
  
```

Semantics of extensions

O_1 flattenable $\llbracket O_1 \text{ then } O_2 \rrbracket_\Gamma = (I, \Sigma', \Psi')$

where

- $\llbracket O_1 \rrbracket_\Gamma = (I, \Sigma_1, \Psi_1)$
- $\llbracket O_2 \rrbracket_{\text{basic}} = (I, \Sigma_2, \Psi_2)$
- $\Sigma' = \Sigma_1 \cup \Sigma_2, \iota : \Sigma_1 \hookrightarrow \Sigma'$
- $\Psi = \iota(\Psi_1) \cup \Psi_2$

O_1 elusive $\llbracket O_1 \text{ then } O_2 \rrbracket_\Gamma = (I, \Sigma', \mathcal{M}')$

where

- $\llbracket O_1 \rrbracket_\Gamma = (I, \Sigma_1, \mathcal{M}_1)$
- $\llbracket O_2 \rrbracket_{\text{basic}} = (I, \Sigma_2, \Psi_2)$
- $\Sigma' = \Sigma_1 \cup \Sigma_2, \iota : \Sigma_1 \hookrightarrow \Sigma'$
- $\mathcal{M}' = \{M \in \mathbf{Mod}(\Sigma') \mid M \models \Psi_2, M|_\iota \in \mathcal{M}_1\}$

Semantics of extensions (cont'd)

`%mcons` (`%def`, `%mono`) leads to the additional requirement that each model in \mathcal{M}_1 has a (unique, unique up to isomorphism) ι -expansion to a model in \mathcal{M}' .

References to Named OSMs

- **Reference** to an OSM existing on the Web
- written directly as a **URL** (or IRI)
- **Prefixing** may be used for abbreviation

`http://owl.cs.manchester.ac.uk/co-ode-files/
ontologies/pizza.owl`

`co-ode:pizza.owl`

Semantics Reference to Named OSMs: $\llbracket iri \rrbracket_{\Gamma} = \Gamma(iri)$

Unions

- O_1 **and** O_2 : union of two stand-alone OSMs
(for extensions O_2 needs to be basic)
- Signatures (and axioms) are **united**
- model classes are **intersected**

`algebra:Monoid` **and** `algebra:Commutative`

Semantics of unions

O_1, O_2 flattenable $\llbracket O_1 \text{ and } O_2 \rrbracket_\Gamma = (I, \Sigma, \Psi)$, where

- $\llbracket O_i \rrbracket_\Gamma = (I, \Sigma_i, \Psi_i)$ ($i = 1, 2$)
- $\Sigma = \Sigma_1 \cup \Sigma_2$, with $\iota_i : \Sigma_i \hookrightarrow \Sigma$
- $\Psi = \iota_1(\Psi_1) \cup \iota_2(\Psi_2)$

one of O_1, O_2 elusive $\llbracket O_1 \text{ and } O_2 \rrbracket_\Gamma = (I, \Sigma, \mathcal{M})$, where

- $\llbracket O_i \rrbracket_\Gamma = (I, \Sigma_i, \mathcal{M}_i)$ ($i = 1, 2$)
- $\Sigma = \Sigma_1 \cup \Sigma_2$, with $\iota_i : \Sigma_i \hookrightarrow \Sigma$
- $\mathcal{M} = \{M \in \mathbf{Mod}(\Sigma) \mid M|_{\iota_i} \in \mathcal{M}_i, i = 1, 2\}$

Translations

- **O with σ** , where σ is a signature morphism
- **O with translation ρ** , where ρ is an **institution comorphism**

ObjectProperty: isProperPartOf

Characteristics: Asymmetric

SubPropertyOf: isPartOf

with translation trans:SR0IQtoCL

then

```
(if (and (isProperPartOf x y) (isProperPartOf y z))
    (isProperPartOf x z))
```

%% transitivity; can't be expressed in OWL together

%% with asymmetry

Semantics of translations

O flattenable Let $\llbracket O \rrbracket_{\Gamma} = (I, \Sigma, \Psi)$

- homogeneous translation
 $\llbracket O \text{ with } \sigma : \Sigma \rightarrow \Sigma' \rrbracket_{\Gamma} = (I, \Sigma', \sigma(\Psi))$
- heterogeneous translation
 $\llbracket O \text{ with translation } \rho : I \rightarrow I' \rrbracket_{\Gamma} = (I', \rho^{Sig}(\Sigma), \rho^{Sen}(\Psi))$

O elusive Let $\llbracket O \rrbracket_{\Gamma} = (I, \Sigma, \mathcal{M})$

- homogeneous translation
 $\llbracket O \text{ with } \sigma : \Sigma \rightarrow \Sigma' \rrbracket_{\Gamma} = (I, \Sigma', \mathcal{M}')$
 where $\mathcal{M}' = \{M \in \mathbf{Mod}(\Sigma') \mid M|_{\sigma} \in \mathcal{M}\}$
- heterogeneous translation
 $\llbracket O \text{ with translation } \rho : I \rightarrow I' \rrbracket_{\Gamma} = (I', \rho^{Sig}(\Sigma), \mathcal{M}')$ where
 $\mathcal{M}' = \{M \in \mathbf{Mod}^{I'}(\rho^{Sig}(\Sigma)) \mid \rho^{Mod}(M) \in \mathcal{M}\}$

Extract – Forget – Hide

	remove/extract	forget/keep	hide/reveal
semantic background	conservative extension	uniform interpolation	model reduct
relation to original	subtheory	interpretable	interpretable
approach	theory level	theory level	model level
type of ontology	flattenable	flattenable	elusive
signature of result	$\geq \Sigma$	$= \Sigma$	$= \Sigma$
change of logic	not possible	possible	possible

Reductions

- intuition: some logical or non-logical symbols are hidden, but the semantic effect of sentences (also those involving these symbols) is kept
- **O reveal Σ** , where Σ is a subsignature of that of O
- **O hide Σ** , where Σ is a subsignature of that of O
- **O hide along μ** , where μ is an **institution morphism**

sort Elem

ops 0:Elem; __+__:Elem*Elem->Elem; inv:Elem->Elem

forall x, y, z . 0+x=x

. $x+(y+z) = (x+y)+z$

. $x+\text{inv}(x)=0$

hide inv

Semantics of reductions

Let $\llbracket O \rrbracket_{\Gamma} = (I, \Sigma, \mathcal{M})$

- homogeneous reduction, reveal
 $\llbracket O \text{ reveal } \Sigma' \rrbracket_{\Gamma} = (I, \Sigma', \mathcal{M}')$
 where $\mathcal{M}' = \{M|_{\iota} \mid M \in \mathcal{M}\}$
 and $\iota : \Sigma' \rightarrow \Sigma$ is the inclusion
- homogeneous reduction, hide
 $\llbracket O \text{ hide } \Sigma' \rrbracket_{\Gamma} = (I, \Sigma \setminus \Sigma', \mathcal{M}')$
 where $\mathcal{M}' = \{M|_{\iota} \mid M \in \mathcal{M}\}$
 and $\iota : \Sigma \setminus \Sigma' \rightarrow \Sigma$ is the inclusion
- heterogeneous reduction
 $\llbracket O \text{ hide along } \rho : I \rightarrow I' \rrbracket_{\Gamma} = (I', \rho^{Sig}(\Sigma), \mathcal{M}')$
 where $\mathcal{M}' = \{\rho^{Mod}(M) \mid M \in \mathcal{M}\}$

Interpolation

- **O forget Σ** , where Σ is a subsignature of that of O
- **O forget Σ with I** , where Σ is a subsignature of that of O , and I is a substitution of that of O
 - intuition: theory of O is interpolated in smaller signature/logic

sort Elem

ops 0:Elem; $++$:Elem*Elem->Elem; inv:Elem->Elem

forall x, y, z . 0+x=x

. $x+(y+z) = (x+y)+z$

. $x+\text{inv}(x)=0$

forget inv

Semantics of interpolations

Note: O must be flattenable!

Let $\llbracket O \rrbracket_{\Gamma} = (I, \Sigma, \Psi)$.

- homogeneous interpolation

$$\llbracket O \text{ keep in } \Sigma' \rrbracket_{\Gamma} = (I, \Sigma', \Psi')$$

where $\Sigma' \subseteq \Sigma$, and $\Psi' \subseteq \mathbf{Sen}(\Sigma')$ is **maximal** with $\Psi \models \iota(\Psi')$,
and $\iota : \Sigma' \rightarrow \Sigma$ is the inclusion

- $\llbracket O \text{ forget } \Sigma' \rrbracket_{\Gamma} = \llbracket O \text{ keep in } \Sigma \setminus \Sigma' \rrbracket_{\Gamma}$

- heterogeneous interpolation

$$\llbracket O \text{ keep in } \Sigma' \text{ with } I' \rrbracket_{\Gamma} = (I', \Sigma', \Psi')$$

where $\rho : I' \rightarrow I$ is the inclusion

and Σ' is such that $\rho^{Sig}(\Sigma') \subseteq \Sigma$

and $\iota : \rho^{Sig}(\Sigma') \rightarrow \Sigma$ is the inclusion

and $\Psi' \subseteq \mathbf{Sen}'(\Sigma')$ is **maximal** with $\Psi \models \iota(\rho^{Sen}(\Psi'))$.

- $\llbracket O \text{ forget } \Sigma' \text{ with } I' \rrbracket_{\Gamma} = \llbracket O \text{ keep in } \Sigma \setminus \Sigma' \text{ with } I' \rrbracket_{\Gamma}$

Module Extractions

- O **extract** $c \Sigma$ **with** m
- Σ : restriction signature (subsignature of that of O)
- c : one of %mcons and %ccons
- m : module extraction method

O must be a conservative extension of the resulting extracted module.

```
co-ode:Pizza extract %mcons
Class: VegetarianPizza
Class: VegetableTopping
ObjectProperty: hasTopping
with locality
```

- Dually: O **remove** $c \Sigma$ **with** m

Semantics of module extractions

Note: O must be flattenable!

Let $\llbracket O \rrbracket_{\Gamma} = (I, \Sigma, \Psi)$.

$\llbracket O \text{ extract } c \Sigma_1 \text{ with } m \rrbracket_{\Gamma} = (I, \Sigma_2, \Psi_2)$

where $(\Sigma_2, \Psi_2) \subseteq (\Sigma, \Psi)$ is computed with m such that

- $\Sigma_2 \supseteq \Sigma_1$
- (Σ, Ψ) is a c -conservative extension of (Σ_2, Ψ_2)

$\llbracket O \text{ remove } c \Sigma_1 \text{ with } m \rrbracket_{\Gamma} = \llbracket O \text{ extract } c \Sigma \setminus \Sigma_1 \text{ with } m \rrbracket_{\Gamma}$

Extract – Forget – Hide

	remove/extract	forget/keep	hide/reveal
semantic background	conservative extension	uniform interpolation	model reduct
relation to original	subtheory	interpretable	interpretable
approach	theory level	theory level	model level
type of ontology	flattenable	flattenable	elusive
signature of result	$\geq \Sigma$	$= \Sigma$	$= \Sigma$
change of logic	not possible	possible	possible

Minimizations (circumscription)

- O_1 then minimize $\{ O_2 \}$
- forces minimal interpretation of non-logical symbols in O_2

Class: Block

Individual: B1 Types: Block

Individual: B2 Types: Block DifferentFrom: B1

then minimize {

Class: Abnormal

Individual: B1 Types: Abnormal }

then

Class: Ontable

Class: BlockNotAbnormal **EquivalentTo:**

Block **and** not Abnormal **SubClassOf:** Ontable

then %implied

Individual: B2 Types: Ontable

Semantics of minimizations

Let $\llbracket O_1 \rrbracket_\Gamma = (I, \Sigma_1, \mathcal{M}_1)$

Let $\llbracket O_1 \text{ then } O_2 \rrbracket_\Gamma = (I, \Sigma_2, \mathcal{M}_2)$

Let $\iota : \Sigma_1 \rightarrow \Sigma_2$ be the inclusion

Then

$$\llbracket O_1 \text{ then minimize } O_2 \rrbracket_\Gamma = (I, \Sigma_2, \mathcal{M})$$

where $\mathcal{M} = \{M \in \mathcal{M}_2 \mid M \text{ is minimal in } \{M' \in \mathcal{M}_2 \mid M'|_\iota = M|_\iota\}\}$

Dually: maximization.

Freeness

- O_1 then free $\{ O_2 \}$
- forces initial interpretation of non-logical symbols in O_2

```

sort Elem
then free {
  sort Bag
  ops mt:Bag;
    __union__:Bag*Bag->Bag, assoc, comm, unit mt
  }

```

Cofreeness

- O_1 then cofree $\{ O_2 \}$
- forces final interpretation of non-logical symbols in O_2

```
sort Elem
then cofree {
  sort Stream
  ops head:Stream->Elem;
      tail:Stream->Stream
}
```


Semantics of freeness

Let $\llbracket O_1 \rrbracket_\Gamma = (I, \Sigma_1, \mathcal{M}_1)$

Let $\llbracket O_1 \text{ then } O_2 \rrbracket_\Gamma = (I, \Sigma_2, \mathcal{M}_2)$

Let $\iota : \Sigma_1 \rightarrow \Sigma_2$ be the inclusion

Then

$$\llbracket O_1 \text{ then free } O_2 \rrbracket_\Gamma = (I, \Sigma_2, \mathcal{M})$$

where $\mathcal{M} = \{M \in \mathcal{M}_2 \mid M \text{ is } Mod(\iota)\text{-free over } M|_\iota \text{ with unit } id\}$

Given a functor $G : \mathbf{B} \rightarrow \mathbf{A}$, an object $B \in \mathbf{B}$ is called *G-free (with unit $\eta_A : A \rightarrow G(B)$) over $A \in \mathbf{A}$* , if for any object $B' \in \mathbf{B}$ and any morphism $h : A \rightarrow G(B')$, there is a unique morphism $h^\# : B \rightarrow B'$ such that $\eta_A; G(h^\#) = h$.

$$\begin{array}{ccc}
 A & \xrightarrow{\eta_A} & G(B) \\
 & \searrow h & \swarrow G(h^\#) \\
 & G(B') &
 \end{array}$$

Semantics of cofreeness

Let $\llbracket O_1 \rrbracket_\Gamma = (I, \Sigma_1, \mathcal{M}_1)$

Let $\llbracket O_1 \text{ then } O_2 \rrbracket_\Gamma = (I, \Sigma_2, \mathcal{M}_2)$

Let $\iota : \Sigma_1 \rightarrow \Sigma_2$ be the inclusion

Then

$$\llbracket O_1 \text{ then cofree } O_2 \rrbracket_\Gamma = (I, \Sigma_2, \mathcal{M})$$

$$\mathcal{M} = \{M \in \mathcal{M}_2 \mid M \text{ is } Mod(\iota)\text{-cofree over } M|_\iota \text{ with counit } id\}$$

Given a functor $G : \mathbf{B} \rightarrow \mathbf{A}$, an object $B \in \mathbf{B}$ is called *G-cofree* (with counit $\varepsilon_A : G(B) \rightarrow A$) over $A \in \mathbf{A}$, if for any object $B' \in \mathbf{B}$ and any morphism $h : G(B') \rightarrow A$, there is a unique morphism $h^\# : B' \rightarrow B$ such that $G(h^\#); \varepsilon_A = h$.

$$\begin{array}{ccc}
 A & \xleftarrow{\varepsilon_A} & G(B) \\
 & \nwarrow h & \nearrow G(h^\#) \\
 & G(B') &
 \end{array}$$

Distributed OSMs

OSM definitions

- **OSM *IRI* = *O* end**
- assigns name *IRI* to OSM *O*, for later reference:

$$\Gamma(IRI) := \llbracket O \rrbracket_{\Gamma}$$

```
ontology co-code:Pizza =
  Class: VegetarianPizza
  Class: VegetableTopping
  ObjectProperty: hasTopping
  . . .
end
```

Interpretations

- **interpretation** $Id : O_1$ to $O_2 = \sigma$
- σ is a signature morphism or a logic translation
- expresses that O_2 logically implies $\sigma(O_1)$

interpretation $i : \text{TotalOrder to Nat} = \text{Elem} \rightarrow \text{Nat}$

interpretation geometry_of_time %mcons :

% Interpretation of linearly ordered time intervals..

int:owltime_le

% ... that begin and end with an instant as lines

% that are incident with linearly ...

to { ord:linear_ordering **and** bi:complete_graphical

% ... ordered points in a special geometry, ...

and int:mappings/owltime_interval_reduction }

= ProperInterval \rightarrow Interval **end**

Semantics of interpretations

Let $\llbracket O_i \rrbracket_\Gamma = (I, \Sigma_i, \mathcal{M}_i)$ ($i = 1, 2$)

$\llbracket \text{interpretation } IRI : O_1 \text{ to } O_2 = \sigma \rrbracket_\Gamma$

is defined iff

$$Mod(\sigma)(\mathcal{M}_2) \subseteq \mathcal{M}_1$$

In this case, $\Gamma(IRI) := ((I, \Sigma_1, \mathcal{M}_1), (I, \Sigma_2, \mathcal{M}_2), \sigma)$.

Equivalences

- **equivalence** $Id : O_1 \leftrightarrow O_2 = O_3$
- expresses that O_1 and O_2 have model classes that are in bijective correspondence
- (fragment) OSM O_3 is such that O_i **then** O_3 is a definitional extension of O_i for $i = 1, 2$.

```
equivalence e : algebra:BooleanAlgebra
               to algebra:BooleanRing =
  <define missing symbols>
end
```

Module Relations

- **module** *Id* $c : O_1$ **of** O_2 **for** Σ
- O_1 is a module of O_2 with restriction signature Σ and conservativity c
 - $c = \%mcons$ every Σ -reduct of an O_1 -model can be expanded to an O_2 -model
 - $c = \%ccons$ every Σ -sentence φ following from O_1 already follows from O_2

This relation shall hold for any module O_1 extracted from O_2 using the **extract** construct.

Alignments

- **alignment** *ld card₁ card₂ : O₁ to O₂ = c₁, ... c_n*
- *card_i* is (optionally) one of 1, ?, +, *
- the *c_i* are correspondences of form *sym₁ rel conf sym₂*
 - *sym_i* is a symbol from *O_i*
 - *rel* is one of >, <, =, %, ∃, ∈, ↦, or an *ld*
 - *conf* is an (optional) confidence value between 0 and 1

Syntax of alignments follows the **alignment API**

<http://alignapi.gforge.inria.fr>

```
alignment Alignment1 : { Class: Woman } to { Class: Person } =
  Woman < Person
end
```

Alignment: Another Example

```
ontology Onto1 =  
  Class: Person  
  Class: Woman SubClassOf: Person  
  Class: Bank  
end
```

```
ontology Onto2 =  
  Class: HumanBeing  
  Class: Woman SubClassOf: HumanBeing  
  Class: Bank  
end
```

```
alignment VAlignment : Onto1 to Onto2 =  
  Person = HumanBeing,  
  Woman = Woman  
end
```

Combinations

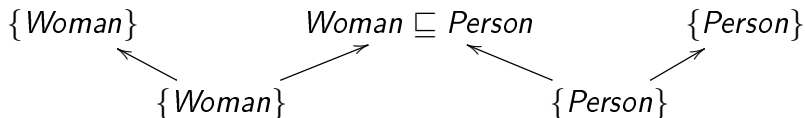
- **combine** O_1, \dots, O_n L_1, \dots, L_m
- L_j are **links** (interpretations, alignments) between OSMs
- The individual OSMs can be prefixed with labels, like $n : O$
- semantics is a **colimit**

```
ontology AlignedOntology1 =
  combine Alignment1
```

```
ontology VAlignedOntology =
  combine 1 : Onto1, 2 : Onto2, VAlignment
  %% 1:Person is identified with 2:HumanBeing
  %% 1:Woman is identified with 2:Woman
  %% 1:Bank and 2:Bank are kept distinct
```

```
ontology VAlignedOntologyRenamed =
  VAlignedOntology with 1:Bank |-> RiverBank,
    2:Bank |-> FinancialBank, Person_HumanBeing |-> Person
```

Diagram for First Alignment



Colimit for First Alignment

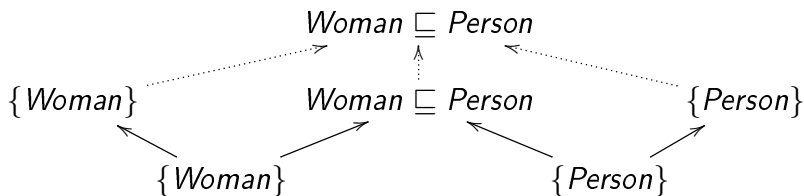
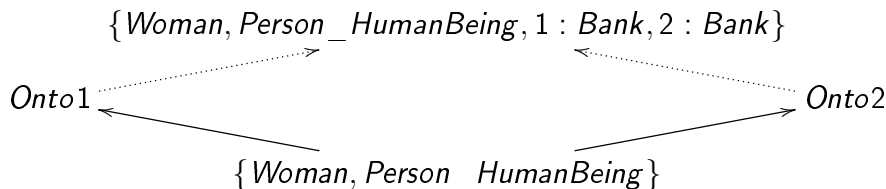


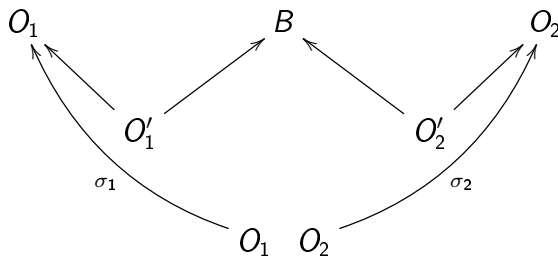
Diagram for Second Alignment



Colimit for Second Alignment



Construction of Diagrams



- $O_1_O_2$ contains, for each $s_1 = s_2$ in A , a symbol $s_1_s_2$
- O'_1 and O'_2 contain the symbols of O_1 and O_2 , respectively, which appear in A in a correspondence $s_1 R s_2$ such that R is not equivalence and B is an OSM constructed
- the signature morphisms σ_1 and σ_2 map each symbol $s_1_s_2$ to s_1 and respectively s_2 .

OSM Bridges

- O_1 bridge with translation t O_2
- t is a logic translation
- semantics: O_1 with translation t then O_2
- t will e.g. translate OWL to some DDL or \mathcal{E} -connections
- O_2 : axioms involving the relations (introduced by t) between OSMs in O_1 .

OSM Bridge Example

```
ontology Publications1 =  
  Class: Publication  
  Class: Article SubClassOf: Publication  
  Class: InBook SubClassOf: Publication  
  Class: Thesis SubClassOf: Publication  
  ...
```

```
ontology Publications2 =  
  Class: Thing  
  Class: Article SubClassOf: Thing  
  Class: BookArticle SubClassOf: Thing  
  Class: Publication SubClassOf: Thing  
  Class: Thesis SubClassOf: Thing
```

OSM Bridge Example, cont'd

ontology Publications_Combined =
combine

1 : Publications1 **with** translation OWL2MS-OWL,

2 : Publications2 **with** translation OWL2MS-OWL

%% *implicitly: Article* \mapsto 1:Article ...

%% *Article* \mapsto 2:Article ...

bridge with translation MS-OWL2DDL

%% *implicitly added my translation MS-OWL2DDL: binary*

1:Publication $\xrightarrow{\sqsubseteq}$ 2:Publication

1:PhdThesis $\xrightarrow{\sqsubseteq}$ 2:Thesis

1:InBook $\xrightarrow{\sqsubseteq}$ 2:BookArticle

1:Article $\xrightarrow{\sqsubseteq}$ 2:Article

1:Article $\xrightarrow{\sqsupseteq}$ 2:Article

Qualifications

Qualifications choose the logic, OSM language and/or serialization:

- **language /**
- **logic /**
- **serialization s**

This affects the subsequent definitions in the distributed OSM.

Alternative Semantics

Institutes

An **institute** $\mathcal{I} = (Sig, \leq, Sen, Mod, \models)$ consists of

- a class Sen of **sentences**;
- a partially ordered class (Sig, \leq) of **signatures**;
- a function $sig : Sen \rightarrow Sig$, giving the (minimal) signature of a sentence (then for each signature Σ , let $Sen(\Sigma) = \{\varphi \in Sen \mid sig(\varphi) \leq \Sigma\}$);
- for each signature Σ , a partially ordered class $Mod(\Sigma)$ of **Σ -models**;
- for each signature Σ , a **satisfaction relation** $\models_{\Sigma} \subseteq Mod(\Sigma) \times Sen(\Sigma)$;
- for any Σ_2 -model M , a Σ_1 -model $M|_{\Sigma_1}$ (called the **reduct**), provided that $\Sigma_1 \leq \Sigma_2$,

Institutes (cont'd)

... such that the following properties hold:

- given $\Sigma_1 \leq \Sigma_2$, for any Σ_2 -model M and any Σ_1 -sentence φ

$$M \models \varphi \text{ iff } M|_{\Sigma_1} \models \varphi$$

(satisfaction is **invariant under reduct**),

- for any Σ -model, $M|_{\Sigma} = M$, and given $\Sigma_1 \leq \Sigma_2 \leq \Sigma$,

$$(M|_{\Sigma_2})|_{\Sigma_1} = M|_{\Sigma_1}$$

(**reducts are compositional**), and

- for any signatures $\Sigma' \leq \Sigma$, and Σ -models $M_1 \leq M_2$, we have $M_1|_{\Sigma'} \leq M_2|_{\Sigma'}$ (**reducts preserve the model ordering**).

Relating Institutes and Institutions

ColInclns = inclusive institutions and comorphisms

ColInstitute = institutes and comorphisms

Theorem

There are functors $G : \text{ColInclns} \rightarrow \text{ColInstitute}$ and $F : \text{ColInstitute} \rightarrow \text{ColInclns}$, such that $G \circ F = \text{id}$

Theorem

For a restriction of DOL, the institute-based and the institution-based semantics coincide (via F and G).

Conclusion

Challenges

- What is a suitable abstract meta framework for **non-monotonic** logics and **rule languages** like RIF and RuleML? Are institutions suitable here? different from those for OWL?
- What is a useful abstract notion of **query** (language) and **answer substitution**?
- How to integrate TBox-like and ABox-like OSMs?
- Can the notions of **class hierarchy** and of **satisfiability** of a class be **generalised** from OWL to other languages?
- How to interpret alignment correspondences with confidence other than 1 in a combination?
- Can **logical frameworks** be used for the specification of OSM languages and translations?
- **Proof support**

Tool support: Heterogeneous Tool Set (Hets)

- available at `hets.dfki.de`
- speaks DOL, HetCASL, CoCASL, CspCASL, MOF, QVT, OWL, Common Logic, and other languages
- analysis
- computation of colimits
- management of proof obligations
- interfaces to theorem provers, model checkers, model finders

Tool support: Ontohub web portal and repository

Ontohub is a web-based repository engine for distributed heterogeneous (multi-language) OSMs

- prototype available at `ontohub.org`
- speaks DOL, OWL, Common Logic, and other languages
- mid-term goal: follow the Open Ontology Repository Initiative (OOR) architecture and API
- API is discussed at `https://github.com/ontohub/OOR_Ontohub_API`
- annual Ontology summit as a venue for review, and discussion

Conclusion

- DOL is a **meta language** for (formal) ontologies, specifications and models (**OSMs**)
- DOL covers many aspects of modularity of and relations among OSMs ("**OSM-in-the large**")
- DOL will be submitted to the OMG as an answer to the **OntoOp** RFP
- **you** can help with joining the **OntoOp** discussion
 - see ontoiop.org