Three Semantics for the Core of the Distributed Ontology Language (Extended Abstract)*

Till Mossakowski[1], **Christoph Lange**[2], **Oliver Kutz**[3]
1. Research Center on Spatial Cognition, University of Bremen and DFKI GmbH Bremen, Germany
2. University of Birmingham, UK
3. Research Center on Spatial Cognition, University of Bremen, Germany
ontoiop-forum@ontolog.cim3.net

Abstract

The Distributed Ontology Language DOL, currently being standardized as ISO WD 17347 within the OntoIOp (Ontology Integration and Interoperability) activity of ISO/TC 37, provides a unified framework for (1) ontologies formalized in heterogeneous logics, (2) modular ontologies, (3) links between ontologies, and (4) ontology annotation.

A DOL ontology consists of modules formalized in languages such as OWL or Common Logic, serialized in the existing syntaxes of these languages. On top, DOL's meta level allows for expressing heterogeneous ontologies and links between ontologies, including (heterogeneous) imports and alignments, conservative extensions, and theory interpretations. We present the abstract syntax of these meta-level constructs, with three alternative semantics: *direct*, *translational*, and *collapsed* semantics.

1 Introduction

A variety of languages is used for formalizing ontologies.[1] OWL is widely used but it restriction to a decidable description logic hinders ontology designers from expressing richer knowledge structures. In biomedical ontologies, for example, mereological relations such as parthood are of great importance, but OWL cannot fully express them. Instead of continuing the earlier practice of embedding first-order mereology axioms as informal annotations into OWL ontologies, we aim at a formal semantics for heterogeneous ontologies. Acknowledging the pluralism in ontology languages, we do not aim at yet another "Esperanto" ontology language, but provide a formal semantics to integrate ontologies written in coexisting different formalisms. We particularly support the widely used OWL, the first-order Common Logic (CL) language, but our framework specifies general conformance criteria and is thus open for plugging in any other ontology language with a formal semantics.

*The paper on which this extended abstract is based was the recipient of the best paper award at the 2012 FOIS conference [Mossakowski *et al.*, 2012].

[1] We take the formal position that an ontology is a formal theory in a logical language ("ontology language") that some community considers suitable for ontology design.

2 The Distributed Ontology Language DOL

The Distributed Ontology Language DOL allows users to use their preferred ontology formalism while becoming interoperable with other formalisms. DOL is the core of the OntoIOp ISO standardization effort on ontology interoperability [OntoIOp, 2012]. It builds on a graph of ontology languages and translations, which will enable users to distribute ontologies by (i) relating ontologies written in different formalisms (e.g. stating that the DOLCE Lite OWL ontology is entailed by the reference first-order DOLCE ontology [Masolo *et al.*, 2003]), (ii) re-using ontology modules given in a different formalism, and (iii) re-using tools such as theorem provers and module extractors along translations between formalisms. (iv) DOL uses IRIs for globally unique identification, which allows for distributing ontologies over the Web. On a meta level on top of basic ontology languages, DOL allows for modeling logically heterogeneous ontologies, comprising of modules written in ontology languages with different underlying logics, and for links between ontologies, such as alignments, relative interpretations or conservative extensions.

We are particularly interested in establishing conformance of the following widely used ontology languages and logics with DOL (ordered by increasing complexity): propositional logic, the W3C-standardized OWL 2 [W3C, 2009] (with its profiles, i.e. sublanguages, EL, RL and QL), classical first-order logic with equality ($\mathsf{FOL}^=$) and the ISO/IEC standard CL [ISO/IEC, 2007], a variant of first-order logic with an impredicative Lisp-like wild-west syntax, coming in the full variant CL (with sequence markers denoting lists of individuals) and the unofficial sublanguage CL^- (without these). Translations between these languages have been developed in previous research [Mossakowski and Kutz, 2011]. Among these languages, CL is most expressive, and therefore can serve as a common translation target. Reasoning about heterogeneous distributed ontologies is possible on demand by translating all modules to first-order logic. This is different from translating them to FOL in the first place, which would no longer give access to optimized language-specific tools decidable/tractable OWL reasoning procedures.

3 An Introductory Example

While mereological relations such as parthood are frequently used in ontologies (including large biomedical ontologies implemented in OWL EL for efficiency), their complete definitions require at least first-order logic and thus exceed the expressiveness of many ontology languages. The following listing shows a heterogeneous mereological ontology, based on an example from [Kutz *et al.*, 2010]:

```
%prefix( %% IRI prefix for this distributed ontology
 : <http://www.example.org/mereology#>
 owl:     <http://www.w3.org/2002/07/owl#>                     %% OWL
 log:     <http://purl.net/dol/logics/>         %% DOL-conforming logics
 trans:   <http://purl.net/dol/translations/>           %% translations
 %% serializations, i.e. concrete syntaxes
 ser:     <http://purl.net/dol/serializations/> %)

distributed-ontology Mereology

%% some basics in propositional logic, so we can use SAT solvers for
%% efficient and early detection of modelling errors
logic log:Propositional syntax ser:Prop/MyNonStandardSyntax
ontology Taxonomy =
 %% basic taxonomic information about mereology reused from DOLCE
 props PT %[ Particular ]%, PD %[ Perdurant ]%,
  T %[ TimeInterval ]%, S  %[ SpaceRegion ]%, AR %[ AbstractRegion ]%
 . S ∨ T ∨ AR ∨ PD ⟶ PT                          %% PT is the top concept
 . S ∧ T ⟶ ⊥                          %% PD, S, T, AR are pairwise disjoint
 . T ∧ AR ⟶ ⊥ %% ...

logic log:SROIQ syntax ser:OWL2/Manchester       %% Parthood in OWL DL,
ontology BasicParthood =                    %%  as far as easily expressible
 Class: ParticularCategory SubClassOf: Particular
  %% similar declarations of the other classes omitted
  DisjointUnionOf: SpaceRegion, TimeInterval, AbstractRegion, Perdurant
   %% pairwise disjointness more compact thanks to an OWL built-in
 ObjectProperty: isPartOf Characteristics: Transitive
 ObjectProperty: isProperPartOf SubPropertyOf: isPartOf
  Characteristics: Asymmetric
 Class: Atom EquivalentTo: inverse isProperPartOf only owl:Nothing

%% The OWL ontology interprets the propositional ontology as follows:
interpretation TaxonomyToParthood : Taxonomy to BasicParthood =
 with translation trans:PropositionalToSROIQ,     %% 1. translate logic
 PT ↦ Particular, S ↦ SpaceRegion,%% 2. rename symbols
 T ↦ TimeInterval, A ↦ AbstractRegion, %[ and so on ]%

logic log:CommonLogic syntax ser:CommonLogic/CLIF           %% Lisp-like
ontology ClassicalExtensionalParthood =    %% import OWL ontology above,
 BasicParthood with translation trans:SROIQtoCL      %% translate it ...
 then {           %% ... to CL, then extend it there, using second-order
         %% features to quantify over the taxonomic category predicates:
  (forall (Cat) (if (or (= Cat S) (= Cat T) (= Cat AR) (= Cat PD))
                  (forall (x y z) (if (and (Cat x) (Cat y) (Cat z))
                                   (and    %% now list all the axioms
   (if (and (isPartOf x y) (isPartOf y x)) (= x y))      %% antisymmetry
   (if (and (isProperPartOf x y)   %% OWL can't express transivitity to-
    (isProperPartOf y z)) (isProperPartOf x z)) %% gether with asymmetry
   (iff (overlaps x y) (exists (pt)
    (and (isPartOf pt x) (isPartOf pt y))))
   (iff (isAtomicPartOf x y) (and (isPartOf x y) (Atom x)))
   (iff (sum z x y) (forall (w) (iff (overlaps w z)  %% existence of sum
    (and (overlaps w x) (overlaps w y))))) (exists (s) (sum s x y)))))))
  (forall (Set a) (iff (fusion Set a) (forall (b) (iff (overlaps b a)
    (exists (c) (and (Set c) (overlaps c a))))))) }  %% 2nd-order fusion
```

4 Syntax of DOL

The following exposition of DOL's syntax and semantics only covers the core meta-logical constructs. A distributed ontology consists of at least one (possibly heterogeneous) **ontology**, plus, optionally, **interpretations** between its participating ontologies. More specifically, a distributed ontology consists of a name, followed by a list of sections, parsed and interpreted subject to the ontology language, logic and syntax[2] set in the section header. Named **ontology definitions** and **links** between ontologies can be items in such a section. An **ontology** O can be, among other cases not covered here, one of the following:

O `::=` $\langle\Sigma,\Delta\rangle$ (a signature Σ and a set of axioms Δ, written in the concrete syntax of some ontology language)
| O **`translate with`** ρ (translation)
| O **`then`** `[ CS ]` $\langle\Sigma,\Delta\rangle$ (extension of one ontology by another basic ontology; optionally marked as conservative)
| `OntoRef` (reference to an ontology on the Web)
| **`logic`** `LogicRef` O (ontology qualified with the logic used to express it; similarly for language and syntax)

An ontology language **translation** ρ is either specified by its name, or it is inferred as the default translation between a given source and target ontology language. A **link** connects two ontologies; possible links include interpretations and alignments. DOL uses IRIs (Internationalized Resource Identifiers, RFC 3987) for names of distributed ontologies, ontologies, logics, translations, links and other objects.

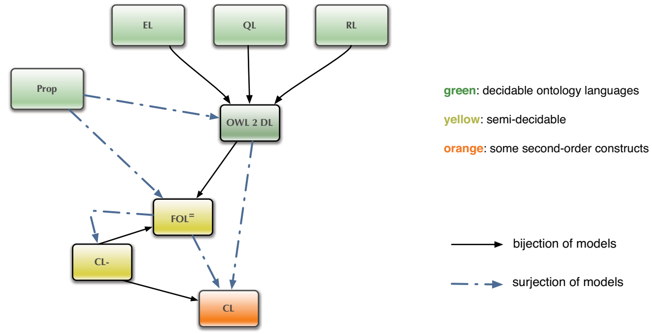

Figure 1: Core of the logic translation graph

5 The Logic Graph

Fig. 1 shows a core fraction of the OntoIOp logic graph, comprising the logics introduced in section 2. An extended graph [Mossakowski and Kutz, 2011; Lange *et al.*, 2012b] also features W3C's RDF and RDFS (following Lucanu *et al.*'s logical conception of them of 2006), relational database schemas [Kutz *et al.*, 2008], distributed description logics [Borgida and Serafini, 2003], $\mathcal{E}$-connections [Kutz *et al.*, 2004], F-logic [Kifer *et al.*, 1995], higher-order logic according to [Borzyszkowski, 1999], CASL [Mosses, 2004], and UML class diagrams [OMG, 2011].

Each of these logics conforms with DOL, which requires notions of *sentence* and *model*, and a relation $\models$ of *satisfaction* between these. DOL provides means for talking about conservative extensions: super-theories that do not introduce new properties (over the *signature*, i.e. the vocabulary, of the old theory). Given signatures $\Sigma_1 \leq \Sigma_2$ (with componentwise inclusion), we assume that all Σ_1-sentences are also Σ_2-sentences, and that each Σ_2 model M_2 has a *reduct* $M_2|_{\Sigma_1}$ to

[2] OWL calls its different syntaxes *serializations*; CL has different *dialects*, which also have different syntaxes.

a Σ_1-model (M_2 is then called an *expansion* of $M_2|_{\Sigma_1}$). For each logic, it is easy to show that $M_2 \models \varphi$ iff $M_2|_{\Sigma_1} \models \varphi$, i.e., satisfaction is invariant under reduct. Finally, we assume a signature union operation.

Fig. 1 also shows some logic translations, with some of their properties. Such a translation consists of three components: a signature translation Φ, which is expected to map signature extensions to signature extensions, a sentence translation α and a model translation β (where models are translated in the reverse direction). Each translation enjoys the following representation condition:

$$\beta(M) \models \varphi \text{ iff } M \models \alpha(\varphi),$$

where φ is a sentence in the source of the translation, and M is a model in the target of the translation. Moreover, we require that each model translation is compatible with reduct, i.e. $\beta(M'|_{\Phi(\Sigma)}) = \beta(M')|_{\Sigma}$ for $\Sigma \leq \Sigma'$ and $M' \in Mod(\Phi(\Sigma'))$. In most cases, model translations will also be surjective. For the translational semantics below, we will need the following notion: We call a translation *weakly exact*, if for each $M' \in Mod(\Sigma')$ and $M_1 \in Mod(\Phi(\Sigma))$ with $\beta(M_1) = M'|_{\Sigma}$, there is a model $M_1' \in Mod(\Phi(\Sigma'))$ with $M_1'|_{\Phi(\Sigma)} = M_1$ and $\beta(M_1') = M'$.

Some of the translations in Fig. 1 are sublogic inclusions (namely the inclusions of the EL, QL and RL into OWL). The other translations generally require some coding. All translations in the graph, except the ones from propositional logic to OWL and the one from OWL to CL are defined to be the *default translations* between their respective logics. Default translations are transitively composable.

6 Semantics of DOL

We pursue a threefold approach of assigning a semantics to the DOL syntax: (i) The **direct model-theoretic semantics uses** the existing semantics of the basic ontology languages, as well as translations between their logics. The semantics of the meta level is specified in semi-formal mathematical textbook style. In this semantics, an ontology denotes a triple $(L, \Sigma, \mathcal{M})$, where L is a logic, Σ a signature in L, and $\mathcal{M}$ a class of Σ-models (ii) The **translational semantics** employs the semantics of CL for all basic ontology languages translatable to CL. The abstract syntax of all basic ontology languages is translated into that of CL, which is then interpreted w.r.t. the CL semantics as in (i). The semantics of the meta level remains semi-formal. (iii) The **collapsed semantics** is fully specified in CL. It translates the abstract syntax of the meta level to CL and then re-uses the CL semantics. Thus, the meta and object levels collapse into CL but may still be distinguished by a closer look into the CL theory.

The model-theoretic nature of the semantics ensures a better representation of the model theory than a theory-level semantics would do. In particular, Theorem 1 ensures that models classes of logical theories represented in CL can be recovered through a model translation. This is of particular importance when studying model-theoretic properties like finite model or tree model properties.

In the context of a global environment Γ, which maps IRIs to (semantics of) ontologies, and the currently selected logic L, the **direct semantics** interprets an ontology O as a signature $\Sigma = sig(\Gamma, L, O)$ in some logic $L' = logic(\Gamma, L, O)$ and a class of models $\mathcal{M} = Mod(\Gamma, L, O)$ over that signature. We combine this into

$$\begin{aligned} sem(\Gamma, L, O) =& (logic(\Gamma, L, O), \\ & sig(\Gamma, L, O), Mod(\Gamma, L, O)). \end{aligned}$$

The direct semantics of a basic ontology is given by its signature and the class of model satisfying its axioms. The semantics of the translation of an ontology along a logic translation uses the signature and model translation components of the logic translation[3]. A reference to a named ontology is looked up in the global environment and, if needed, translated to the current logic (this kind of implicit coercion is used in the listing in section 3 when including an OWL ontology into a CL ontology). Finally, a logic qualification replaces the current logic with a new one.

O'	$sem(\Gamma, L, O') = \ldots$
$\langle \Sigma, \Delta \rangle$	$(L, \Sigma, \{M \in Mod(\Sigma) \mid M \models \Delta\})$
O `translate with` ρ	Let $\Sigma = sig(\Gamma, L, O)$ and $\rho = (\Phi, \alpha, \beta) : L_1 \to L_2$. Then $logic(\Gamma, L, O') = L_2$, $sig(\Gamma, L, O') = \Phi(\Sigma)$, and $Mod(\Gamma, L, O') = \{M \in Mod(\Phi(\Sigma)) \mid \beta(M) \subseteq Mod(\Gamma, L, O)\}$
O `then [CS]` $\langle \Sigma', \Delta' \rangle$	Let $\Sigma = sig(\Gamma, L, O)$. Then $sig(\Gamma, L, O') = \Sigma \cup \Sigma'$ and $Mod(\Gamma, L, O') = \{M' \in Mod(\Sigma \cup \Sigma') \mid M' \models \Delta'$ and $M'|_{\Sigma} \in Mod(\Gamma, L, O)\}$
`OntoRef`	$(L, \Phi(\Sigma), \{M \in Mod(\Phi(\Sigma)) \mid \beta(M) \subseteq \mathcal{M}\}$ where $\Gamma(\texttt{OntoRef}) = (L_1, \Sigma, \mathcal{M})$ and $(\Phi, \alpha, \beta) : L_1 \to L$ is the default translation
`logic LogicRef` O	$sem(\Gamma, \texttt{LogicRef}, O)$

The semantics of relative interpretations is formulated in terms of model class inclusion:

$sem(\Gamma, L, \texttt{interpretation IntprName : } O_1 \texttt{ to } O_2)) =$
$\Gamma[\texttt{IntprName} \mapsto (\Sigma_1, \Sigma_2)]$ where
$(L_1, \Sigma_1, \mathcal{M}_1) = sem(\Gamma, L, O_1)$,
$(L_2, \Sigma_2, \mathcal{M}_2) = sem(\Gamma, L, O_2)$, and the semantics is
defined only if $\beta(\mathcal{M}_2|_{\Sigma_1}) \subseteq \mathcal{M}_1$, where (Φ, α, β) is the
default translation from L_1 to L_2.

With the direct semantics, one can define many standard logical notions in a straightforward way. For example, a heterogeneous ontology is *satisfiable* if it has a nonempty model class. A sentence φ is a *logical consequence* of a heterogeneous ontology O in context Γ and L, written $\Gamma, L, O \models \varphi$, if φ is a sentence in the logic $logic(\Gamma, L, O)$, and each model in $Mod(\Gamma, L, O)$ satisfies φ.

The **translational semantics** uses CL as a foundational framework for the distributed ontology language DOL, similar to what set theory provides for general mathematical theories. This semantics assumes that each involved ontology

[3] A theory-level semantics would use the sentence translation instead of the model translation.

language is mapped to CL by a weakly exact translation. The semantics is defined by first translating a heterogeneous ontology to CL, and then using the direct semantics for the result.

The **collapsed semantics** requires the representation of the meta level within CL. For this purpose, the direct model-level semantics should be complemented by a theory-level semantics: a distributed ontology then denotes a basic theory in some logic (which amounts to flattening out all structure), plus some conditions for conservativity and relative interpretations. For each logic, one needs to axiomatize a specific partial order of signatures in CL, plus a set of sentences equipped with a logical consequence relation. In order to avoid the formalization of models and the satisfaction relation (which would require the inclusion of a set theory like ZFC), a sound and complete calculus is axiomatized for each logic. For each logic translation, the signature and sentence translations need to be axiomatized. We require that this axiomatization is done in such a way that the resulting semantics is compatible with the translational semantics. Although this formalization is doable in principle, we refrain form providing the (massive) details.

7 Relations Among the Different Semantics

Theorem 1 (Compatibility of semantics) *Given an ontology O written in logic L, if O does not involve conservative extensions, then the direct semantics and the translational semantics coincide:*

$$\Phi(\Sigma) = \Sigma_{\mathsf{CL}} \text{ and } \mathcal{M} = \beta(\mathcal{M}_{\mathsf{CL}})$$

where $(L, \Sigma, \mathcal{M})$ *is the direct semantics of O,* $(\mathsf{CL}, \Sigma_{\mathsf{CL}}, \mathcal{M}_{\mathsf{CL}})$ *is the translational semantics of O, and* $\rho = (\Phi, \alpha, \beta)$ *is the default translation from L to* CL.[4]

While theorem 1 provides a good compatibility of the first two semantics, there are some differences:

Example 2 (Conservative extensions) *Conservative extensions, both in the consequence-theoretic and the model-theoretic variant, play an important role, in particular concerning ontology module extraction, see e.g. [Lutz and Wolter, 2010]. Consider a large ontology like SNOMED CT [Schulz* et al.*, 2009]. For a particular application, typically only a small portion, i.e. a subtheory, of the ontology is needed. However, this subtheory should contain all logical information about its signature that is implied by the whole ontology. This requirement is precisely that of conservative extension. Hence, a* ***module*** *is a subtheory of an ontology such that the ontology is a conservative extension of the module. Consider the following* EL *theory about lectures and their subjects [Lutz* et al.*, 2007]:*[5]

$$\mathsf{Lecture} \sqsubseteq \exists \mathsf{has\_subject.Subject} \sqcap \exists \mathsf{given\_by.Lecturer}$$
$$\mathsf{Intro\_AI} \sqsubseteq \mathsf{Lecture}$$

This theory is extended as follows:

$$\mathsf{Intro\_AI} \sqsubseteq \exists \mathsf{has\_subject.Logic}$$
$$\mathsf{Intro\_AI} \sqsubseteq \exists \mathsf{has\_subject.NeuralNetworks}$$
$$\mathsf{Logic} \sqcap \mathsf{NeuralNetworks} \sqsubseteq \bot$$

Now this extended theory logically implies that $\mathsf{Intro\_AI} \sqsubseteq\ \geq 2\, \mathsf{has\_subject}$*; this follows since* Logic *and* NeuralNetworks *are disjoint and both related via* has_subject *to* Intro_AI*. Hence, in* OWL*, the larger theory is not a consequence-theoretic conservative extension of the smaller one, because* $\mathsf{Intro\_AI} \sqsubseteq\ \geq 2\, \mathsf{has\_subject}$ *is a sentence in the signature of the smaller theory that follows from the larger theory, but not from the smaller one. But in* EL*, such a sentence does not exist. In particular, the number restriction* $\geq 2\, \mathsf{has\_subject}$ *cannot be expressed in* EL.

This example shows that the translational semantics differs from the direct semantics w.r.t reasoning tasks that are syntax sensitive. While the direct semantics would recognize the above extension as consequence-theoretically conservative (because it directly works with EL), the translational semantics would not, because after translating to CL, the conservative extension property is lost. A similar difference arises when computing the least common subsumer of a set of concept descriptions, because this also depends on the expressiveness of the used language.

8 Conclusion

We have presented three different semantics for the Distributed Ontology Language DOL, which are compatible with each other, with some exceptions like conservative extensions. The direct semantics stays close to the semantics of the individual ontology languages such as OWL and CL. The translational semantics is based on a mapping to CL, such that only knowledge of CL is required to understand this semantics. The collapsed semantics formalizes also the meta theory in CL, and thus makes the meta level itself amenable to computer-assisted theorem proving and verification.

A first application of DOL is the (indeed homogeneous) COLORE repository [Grüninger,], containing more than 400 theories written in CL. So far, the relations between these theories have been stated informally and proved manually. With DOL, these relations can be stated formally. This allows for using theorem provers and model finders for (dis-)proving logical facts about these relations [Lange *et al.*, 2012a].

Future work will include specification of further logics and translations. The direct semantics has a clear advantage here, because the logics can be directly included in the graph, while the translational semantics first requires translation to CL, which may be involved or impossible.

Acknowledgments: We would like to thank the OntoIOp working group within ISO/TC 37/SC 3 for providing valuable feedback, in particular Michael Grüninger, Pat Hayes, Maria Keet, Chris Menzel, and John Sowa. This work has been supported by the DFG-funded Research Centre on Spatial Cognition (SFB/TR 8), project I1-[OntoSpace], and EPSRC grant EP/J007498/1.

[4]If we take global environments into account, these need to be compatible in an obvious way, see [Mossakowski *et al.*, 2012] for details.

[5]Note that the $\bot$ concept is part of the $\mathcal{EL}$ ++ extension of the logic $\mathcal{EL}$ and so available in the OWL profile EL.

References

[Borgida and Serafini, 2003] A. Borgida and L. Serafini. Distributed Description Logics: Assimilating Information from Peer Sources. *Journal of Data Semantics*, 1:153–184, 2003.

[Borzyszkowski, 1999] Tomasz Borzyszkowski. Higher-order logic and theorem proving for structured specifications. In Didier Bert, Christine Choppy, and Peter D. Mosses, editors, *WADT*, volume 1827 of *Lecture Notes in Computer Science*, pages 401–418. Springer, 1999.

[Grüninger,] Michael Grüninger. COLORE: Common logic ontology repository. http://code.google.com/p/colore/.

[ISO/IEC, 2007] Information technology — Common Logic (CL): a framework for a family of logic-based languages. Technical Report 24707:2007, ISO/IEC, 2007. http://iso-commonlogic.org.

[Kifer *et al.*, 1995] Michael Kifer, Georg Lausen, and James Wu. Logical Foundations of Object-Oriented and Frame-Based Languages. *Journal of the ACM*, 42:741–843, July 1995.

[Kutz *et al.*, 2004] O. Kutz, C. Lutz, F. Wolter, and M. Zakharyaschev. $\mathcal{E}$-Connections of Abstract Description Systems. *Artificial Intelligence*, 156(1):1–73, 2004.

[Kutz *et al.*, 2008] Oliver Kutz, Dominik Lücke, and Till Mossakowski. Heterogeneously Structured Ontologies—Integration, Connection, and Refinement. In Thomas Meyer and Mehmet A. Orgun, editors, *Advances in Ontologies. Proc. of the KR-08 Ontology Workshop (KROW 2008)*, volume 90 of *CRPIT*, pages 41–50, Sydney, Australia, 2008. ACS.

[Kutz *et al.*, 2010] O. Kutz, T. Mossakowski, and D. Lücke. Carnap, Goguen, and the Hyperontologies: Logical Pluralism and Heterogeneous Structuring in Ontology Design. *Logica Universalis*, 4(2):255–333, 2010. Special Issue on 'Is Logic Universal?'.

[Lange *et al.*, 2012a] Christoph Lange, Oliver Kutz, Till Mossakowski, and Michael Grüninger. The distributed ontology language (DOL): Ontology integration and interoperability applied to mathematical formalization. In Johan Jeuring, John A. Campbell, Jacques Carette, Gabriel Dos Reis, Petr Sojka, Makarius Wenzel, and Volker Sorge, editors, *Intelligent Computer Mathematics*, number 7362 in LNAI, pages 463–467. Springer Verlag, 2012.

[Lange *et al.*, 2012b] Christoph Lange, Till Mossakowski, and Oliver Kutz. LoLa: A modular ontology of logics, languages, and translations. In Thomas Schneider and Dirk Walther, editors, *Modular Ontologies*, number 875 in CEUR Workshop Proceedings, Aachen, 2012.

[Lucanu *et al.*, 2006] Dorel Lucanu, Yuan-Fang Li, and Jin Song Dong. Semantic Web Languages—Towards an Institutional Perspective. In Kokichi Futatsugi, Jean-Pierre Jouannaud, and José Meseguer, editors, *Algebra, Meaning, and Computation, Essays Dedicated to Joseph A. Goguen on the Occasion of His 65th Birthday*, volume 4060 of *Lecture Notes in Computer Science*, pages 99–123. Springer, 2006.

[Lutz and Wolter, 2010] C. Lutz and F. Wolter. Deciding inseparability and conservative extensions in the description logic $\mathcal{EL}$. *Journal of Symbolic Computation*, 45(2):194–228, 2010.

[Lutz *et al.*, 2007] C. Lutz, D. Walther, and F. Wolter. Conservative Extensions in Expressive Description Logics. In *Proceedings of IJCAI-07*, pages 453–458. AAAI Press, 2007.

[Masolo *et al.*, 2003] C. Masolo, S. Borgo, A. Gangemi, N. Guarino, and A. Oltramari. WonderWeb Deliverable D18: Ontology Library. Technical report, ISTC-CNR, 2003.

[Mossakowski and Kutz, 2011] T. Mossakowski and O. Kutz. The Onto-Logical Translation Graph. In *Modular Ontologies—5th International Workshop (WoMO)*, volume 230 of *Frontiers in Artificial Intelligence and Applications*, pages 94–109. IOS Press, 2011.

[Mossakowski *et al.*, 2012] Till Mossakowski, Oliver Kutz, and Christoph Lange. Three semantics for the core of the Distributed Ontology Language. In Maureen Donnelly and Giancarlo Guizzardi, editors, *Formal Ontology in Information Systems*, number 239 in Frontiers in Artificial Intelligence and Applications, pages 337–352. IOS Press, 2012.

[Mosses, 2004] Peter D. Mosses, editor. *CASL Reference Manual*. Number 2960 in LNCS. Springer Verlag, 2004.

[OMG, 2011] OMG Unified Modeling Language (UML) superstructure. Technical Report formal/2011-08-06, Object Management Group, August 2011.

[OntoIOp, 2012] OntoIOp (ontology integration and interoperability), the ISO 17347 standard development initiative, 2012. http://ontoiop.org.

[Schulz *et al.*, 2009] S. Schulz, B. Suntisrivaraporn, F. Baader, and M. Boeker. SNOMED reaching its adolescence: Ontologists' and logicians' health check. *Int. J. of Medical Informatics*, 78(Supplement 1):S86–S94, 2009.

[W3C, 2009] OWL 2 web ontology language: Document overview. W3C recommendation, World Wide Web Consortium (W3C), October 2009.