The great table of Description Logics and formal ontology notations

Jean-Baptiste Lamy / jean-baptiste.lamy @ univ-paris13.fr

June 22, 2018

Description Logics (DL) are the logics used to formalize ontology [1]. Many notations are used to express DL, *e.g.* in equations in scientific papers, in editor software like Protégé, or in programming languages. Moreover, the semantics of DL is usually expressed in first-order logics or as set formula. These notations are difficult to understand and to translate from one to another.

This is why I propose here a big table (next page) that compares 5 notations related to DL and formal ontologies:

1. DL syntax (as commonly used in equations and scientific papers)
2. Protégé editor (expression editor syntax)
3. Owlready2 (a package for ontology-oriented programming in Python [2, 3])
4. Semantics in first-order logic
5. Semantics in set formula

This table is an augmented and improved version of the one I presented in [2] and in my habilitation thesis [4].

Background on DL semantics

DL have a model-theoretic semantics, which is defined in terms of interpretations. For a given ontology $\mathcal{O}$, an interpretation $\mathcal{I} = (\Delta, f)$ is a tuple where the domain $\Delta = \{...\}$ is a non-empty set of objects and the interpretation function f is a function that associates each individual i, class A, role R, composed expression (defined with semantic connectors) and axiom with its interpretation over Δ, as follows:

$f(i \in \mathbb{I}) \in \Delta$
$f(A \in \mathbb{C}) \subseteq \Delta$
$f(R \in \mathbb{R}) \subseteq (\Delta \times \Delta)$

Note: f and Δ are sometimes written $\cdot^I$ and Δ^I; in this case $x^I = f(x)$.

References

[1] F Baader, D Calvanese, D L McGuinness, D Nardi, and P L Patel-Schneider. *The description logic handbook: theory, implementation and applications.* Cambridge University Press, 2007.

[2] Lamy JB. Owlready: Ontology-oriented programming in Python with automatic classification and high level constructs for biomedical ontologies. *Artif Intell Med*, 80:11–28, 2017.

[3] Lamy JB. Ontology-Oriented Programming for Biomedical Informatics. *Stud Health Technol Inform*, 221:64–68, 2016.

[4] Lamy JB. *Représentation, iconisation et visualisation des connaissances : Principes et applications à l'aide à la décision médicale.* PhD thesis, Université de Rouen-Normandie, 2017.

Available at:

- http://www.lesfleursdunormal.fr/_downloads/article_owlready_aim_2017.pdf
- http://www.lesfleursdunormal.fr/static/_downloads/hdr.pdf

		DL syntax	Protégé	Python + Owlready2	First-order logic	Semantics in set formula
Const.	Top	$\top$	Thing	Thing	$\top$, such as $\forall x, \top(x) = true$	Δ
	Bottom	$\bot$	Nothing	Nothing	$\bot$, such as $\forall x, \bot(x) = false$	$\emptyset$
Axioms	Subsumption	$A \sqsubseteq B$	A subclass of B	class A(B): ... (assertion) A.is_a.append(B) (assertion) issubclass(A, B) (test)	$\forall x, A(x) \rightarrow B(x)$	$f(A) \subseteq f(B)$
		$R \sqsubseteq S$	R subproperty of S	(same as above)	$\forall x \forall y, R(x,y) \rightarrow S(x,y)$	$f(R) \subseteq f(S)$
	Equivalence	$A \equiv B$	A equivalent to B	A.equivalent_to.append(B) (as.) B in A.equivalent_to (test)	$\forall x, A(x) \leftrightarrow B(x)$	$f(A) = f(B)$
	Instanciation	$A(i)$	i type A	i = A() (assertion) i.is_instance_of.append(A) isinstance(i, A) (test)	$A(i)$	$f(i) \in f(A)$
	Relations	$R(i,j)$	i object property assertion j i data property assertion j	i.R = j (R is functional) i.R.append(j) (otherwise)	$R(i,j)$	$(f(i), f(j)) \in f(R)$
Semantic connectors	Complement	$\neg A$	not A	Not(A)	$\neg A(x)$	$\Delta \backslash f(A)$
	Intersection	$A \sqcap B$	A and B	A & B (or) And([A, B,...])	$A(x) \wedge B(x)$	$f(A) \cap f(B)$
	Union	$A \sqcup B$	A or B	A | B (or) Or([A, B,...])	$A(x) \vee B(x)$	$f(A) \cup f(B)$
	Extension	$i, j, ...$	{i, j,...}	OneOf([i, j,...])	$x \in \{i, j, ...\}$	$\{f(i), f(j), ...\}$
	Inverse	R^{-}	inverse of R	Inverse(R) (construct) S.inverse = R (assertion)	$\forall i \forall j, S(i,j) = R(j,i)$	$\{(a,b) \mid (b,a) \in f(R)\}$
	Transitive closure	R^{+}	-	-		$\cup_{i \geq 1}(f(R))^{i}$
	Composition	$R \circ S$	R o S	PropertyChain([R, S])		$\{(a,c) \in \Delta \times \Delta \mid \exists b, (a,b) \in f(R) \wedge (b,c) \in f(S)\}$
	Existential quantifier	$\exists R.B$	R some B	R.some(B)	$\exists y, R(x,y) \wedge B(y)$	$\{a \in \Delta \mid \exists b, (a,b) \in f(R) \wedge b \in f(B)\}$
	Universal quantifier	$\forall R.B$	R only B	R.only(B)	$\forall y, R(x,y) \rightarrow B(y)$	$\{a \in \Delta \mid \forall b, (a,b) \in f(R) \rightarrow b \in f(B)\}$
	Number restrictions	$= 2R.B$	R exactly 2 B	R.exactly(2, B)	$\lvert\{y \mid R(x,y) \wedge B(y)\}\rvert = 2$	$\{a \in \Delta \mid \lvert\{b \mid (a,b) \in f(R) \wedge b \in f(B)\}\rvert = 2\}$
		$\leq 2R.B$	R max 2 B	R.max(2, B)	$\lvert\{y \mid R(x,y) \wedge B(y)\}\rvert \leq 2$	$\{a \in \Delta \mid \lvert\{b \mid (a,b) \in f(R) \wedge b \in f(B)\}\rvert \leq 2\}$
		$\geq 2R.B$	R min 2 B	R.min(2, B)	$\lvert\{y \mid R(x,y) \wedge B(y)\}\rvert \geq 2$	$\{a \in \Delta \mid \lvert\{b \mid (a,b) \in f(R) \wedge b \in f(B)\}\rvert \geq 2\}$
	Role filler	$\exists R.\{j\}$	R value j	R.value(j)	$R(x,j)$	$\{a \in \Delta \mid (a, f(j)) \in f(R)\}$
Decomposable	Disjoint	$A \sqcap B \sqsubseteq \bot$	A disjoint with B	AllDisjoint([A, B])	$\forall x, \neg(A(x) \wedge B(x))$	$f(A) \cap f(B) = \emptyset$
	Property domain	$\exists R.\top \sqsubseteq A$	R domain A	R.domain = [A]	$\forall x, (\exists y, R(x,y)) \rightarrow A(x)$	$f(R) \subseteq \{(a,b) \mid a \in f(A)\}$
	Property range	$\top \sqsubseteq \forall R.B$	R range B	R.range = [B]	$\forall x \forall y, R(x,y) \rightarrow B(y)$	$f(R) \subseteq \{(a,b) \mid b \in f(B)\}$
	Role filler as class property	$A \sqsubseteq \exists R.\{j\} \wedge (\exists R^{-}.A)(j)$	-	A.R = j (R is functional) A.R.append(j) (otherwise)	-	-
	Local closed world	-	-	close_world(A)	-	-

The great table of Description Logics and formal ontology notations (logics, programming and semantics) —