OWL FA: A Metamodeling Extension of OWL DL$^{(*)}$

Jeff Z. Pan,[1] Ian Horrocks[1] and Guus Schreiber[2]

[1] School of Computer Science, University of Manchester, UK
{pan, horrocks}@cs.man.ac.uk
[2] Computer Science Department, Free University Amsterdam, Netherlands
schreiber@cs.vu.nl

Abstract. Recent research has shown that the semantics of the standard SW annotation language RDF (as well as its ontological extension RDFS) and that of the standard SW ontology language OWL DL are not compatible with each other. Pan and Horrocks [5] propose a sub-language of RDFS, called RDFS(FA), which provides a clear connection between RDF and OWL DL. This paper proposes OWL FA, an extension of OWL DL with the metamodeling architecture of RDFS(FA). It also investigates some reasoning tasks for OWL FA.

1 Introduction

The OWL Web Ontology Language is a W3C recommendation for expressing ontologies in the Semantic Web. There are three sub-language of OWL; namely, OWL Lite, OWL DL and OWL Full. In terms of both semantics and expressive power, OWL Lite and OWL DL are quite similar, while OWL DL and OWL Full are very different. OWL DL is regarded as the *key* OWL, which provides a good balance between expressive power and decidability. OWL Full is thought of as a 'large' OWL, which provides a free mixing of RDF and OWL, although it is clearly undecidable and has a non-standard semantics.

A common user complaint about OWL DL is that it does not allow them to describe additional level(s) of classes and properties. It has been pointed out that if we just ignore the need for meta-classes and meta-properties, some users will simply not use OWL, and the whole effort could become a failure [9].

Example 1. A well known example [10] of using meta-classes and meta-properties is the WordNet ontology: WordNet synsets (the basic WordNet concepts) are represented as instances of the meta-class LexicalConcept. WordNet $hyponym$ relations (the subclass relations in WordNet) are represented as tuples of the meta-property $hyponymOf$ relation between instances of wns:LexicalConcept. These can be represented in the following RDFS statements, but not in OWL DL.

```
wns:LexicalConcept rdfs:subClassOf rdfs:Class .
wns:hyponymOf rdfs:subPropertyOf rdfs:subClassOf;
      rdfs:domain wns:LexicalConcept ;
      rdfs:range wns:LexicalConcept .
wnc:100002086 wns:hyponymOf wnc:100001740 .
```

Furthermore, we want to express that wns:*hyponymOf* is a transitive property:

$$\mathrm{Trans}(\texttt{wns:}\textit{hyponymOf}),$$

so that if

$$\langle \texttt{wnc:}100002086, \texttt{wnc:}100001740\rangle : \texttt{wns:}\textit{hyponymOf}$$
$$\langle \texttt{wnc:}100001740, \texttt{wnc:}100001923\rangle : \texttt{wns:}\textit{hyponymOf},$$

we can entail that wnc:100002086 is a wns:*hyponymOf* wnc:100001923.

$^{(*)}$ This work is partially supported by the FP6 Network of Excellence EU project Knowledge Web (IST-2004-507842).

OWL Full can be regarded as a not so successful attempt at integrating RDF with OWL DL. Firstly, there are at least three known problems in extending the RDF(S) Model Theory (RDF MT) [1] with OWL constructors [6, 7, 2]. Due to these problems, it is unknown whether the OWL Full semantics could give a coherent meaning to OWL Full ontologies; i.e., there may be OWL Full ontologies for which the semantics would not be well defined. Secondly, there is a serious mismatch between the semantics of OWL DL and OWL Full because OWL Full disagrees with OWL DL on valid OWL DL ontologies. More specifically, for two *OWL DL* ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$, $\mathcal{O}_1$ OWL Full-entails $\mathcal{O}_2$ does *not* imply that $\mathcal{O}_1$ OWL DL-entails $\mathcal{O}_2$ [8]. Furthermore, it has been shown that the metamodeling architecture of OWL Full also contributes to its undecidability [3].

In this paper, we propose OWL FA, an extension of OWL DL with the metamodeling architecture of RDFS(FA), which is a sub-language of RDFS that provides a clear connection with OWL DL. Intuitively, RDFS(FA) stratifies the one-layer metamodeling architecture into a multiple-layer metamodeling architecture, so as to overcome the problem of dual roles that RDFS has. The satisfiability-preserving bi-directional mapping between RDFS(FA) axioms in 0-1 strata and OWL DL axioms suggests that we could extend OWL DL with the metamodeling architecture of RDFS(FA) [4].

2 OWL FA

Intuitively, OWL FA introduces a stratum number in class constructors and axioms to indicate the strata they belong to. Let $\mathrm{i} \geq 0$ be an integer. OWL FA consists of an alphabet of distinct class names $\mathbf{V}_{\mathrm{C_i}}$ (for stratum i), datatype names $\mathbf{V}_{\mathrm{D}}$, abstract property names $\mathbf{V}_{\mathrm{AP_i}}$ (for stratum i), datatype property names $\mathbf{V}_{\mathrm{DP}}$ and individual (object) names ($\mathbf{V}_{\mathrm{I}}$); together with a set of constructors (with subscriptions) to construct *OWL FA-classes* and *OWL FA-properties*.

OWL FA has a model theoretic semantics, which is defined in terms of interpretations. Given an OWL FA alphabet $\mathbf{V}$, a set of built-in datatype names $\mathbf{B} \subseteq \mathbf{V}_{\mathrm{D}}$ and an integer $\mathrm{k} \geq 1$, an *OWL FA interpretation* is a pair $\mathcal{J} = (\Delta^{\mathcal{J}}, \cdot^{\mathcal{J}})$, where $\Delta^{\mathcal{J}}$ is the domain (a non-empty set) and $\cdot^{\mathcal{J}}$ is the interpretation function, which satisfy the following conditions (where $0 \leq \mathrm{i} \leq \mathrm{k}$):

1. $\Delta^{\mathcal{J}} = \bigcup_{0 \leq \mathrm{i} \leq \mathrm{k}-1} \Delta_{A_{\mathrm{i}}}^{\mathcal{J}} \cup \Delta_{\mathbf{D}}$, where $\Delta_{A_{\mathrm{i}}}^{\mathcal{J}}$ is the domain for stratum i and $\Delta_{\mathbf{D}}$ is the datatype domain;
2. $\Delta_{A_{\mathrm{i+1}}}^{\mathcal{J}} = 2^{\Delta_{A_{\mathrm{i}}}^{\mathcal{J}}} \cup 2^{\Delta_{A_{\mathrm{i}}}^{\mathcal{J}} \times \Delta_{A_{\mathrm{i}}}^{\mathcal{J}}}$ and $\Delta_{\mathbf{D}} \cap \Delta_{A_{\mathrm{i}}}^{\mathcal{J}} = \emptyset$;
3. $\forall \mathsf{a} \in \mathbf{V}_{\mathrm{I}} : \mathsf{a}^{\mathcal{J}} \in \Delta_{A_0}^{\mathcal{J}}$ and $\forall \mathsf{C} \in \mathbf{V}_{\mathrm{C_{i+1}}} : \mathsf{C}^{\mathcal{J}} \subseteq \Delta_{A_{\mathrm{i}}}^{\mathcal{J}}$;
4. $\forall R \in \mathbf{V}_{\mathrm{AP_{i+1}}} : R^{\mathcal{J}} \subseteq \Delta_{A_{\mathrm{i}}}^{\mathcal{J}} \times \Delta_{A_{\mathrm{i}}}^{\mathcal{J}}$ and $\forall T \in \mathbf{V}_{\mathrm{DP}} : T^{\mathcal{J}} \subseteq \Delta_{A_0}^{\mathcal{J}} \times \Delta_{\mathbf{D}}$;
5. $\bigcup_{\forall d \in \mathbf{B}} V(d) \subseteq \Delta_{\mathbf{D}}$, where $V(d)$ is the value space of d;
6. $\forall d \in \mathbf{V}_{\mathrm{D}}$, if $d \in \mathbf{B}$, then[3]
   (a) $d^{\mathcal{J}} = V(d)$, where $V(d)$ is the value space of d,
   (b) if $v \in L(d)$, then $(\text{“}v\text{”}^{\wedge\wedge}d)^{\mathcal{J}} = L2V(d)(v)$, where $L(d)$ is lexical space of d and $L2V(d)$ is the lexical-to-value mapping of d,
   (c) if $v \notin L(d)$, then $(\text{“}v\text{”}^{\wedge\wedge}d)^{\mathcal{J}}$ is undefined;

   otherwise, $d^{\mathcal{J}} \subseteq \Delta_{\mathbf{D}}$ and $\text{“}v\text{”}^{\wedge\wedge}d \in \Delta_{\mathbf{D}}$.

In the rest of the paper, we assume that i is an integer such that $1 \leq \mathrm{i} \leq \mathrm{k}$. The interpretation function can be extended to give semantics to OWL FA-properties and OWL FA-classes. Let $RN \in \mathbf{V}_{\mathrm{AP_i}}$ be an abstract property name in stratum i and R an abstract property in stratum i. Valid OWL FA abstract properties are defined by the abstract syntax: $R ::= RN \mid R^{-}$, where for some $x, y \in \Delta_{A_{\mathrm{i}-1}}^{\mathcal{J}}$, $\langle x, y \rangle \in R^{\mathcal{J}}$ iff $\langle y, x \rangle \in R^{-\mathcal{J}}$. Valid OWL FA datatype properties are datatype property names.

Now we define the OWL FA-class descriptions. Let $\mathsf{CN} \in \mathbf{V}_{\mathrm{C_i}}$ be an atomic class name in stratum i, R an OWL FA-property in stratum i, $o \in \mathbf{V}_{\mathrm{I}}$ an individual, $T \in \mathbf{V}_{\mathrm{DP}}$ a datatype property name, and C, D OWL FA-classes in stratum i. Valid OWL FA-classes are defined by the abstract syntax:

$$C ::= \top_{\mathrm{i}} \mid \bot \mid \mathsf{CN} \mid \neg_{\mathrm{i}} C \mid C \sqcap_{\mathrm{i}} D \mid C \sqcup_{\mathrm{i}} D \mid \{o\} \mid \exists_{\mathrm{i}} R.C \mid \forall_{\mathrm{i}} R.C \mid \leqslant_{\mathrm{i}} nR \mid \geqslant_{\mathrm{i}} nR$$
$$(\text{if i = 1})\ \exists_1 T.d \mid \forall_1 T.d \mid \leqslant_1 nT \mid \geqslant_1 nT$$

Constructor	DL Syntax	Semantics
top	$\top_{\mathrm{i}}$	$\Delta_{A_{\mathrm{i}-1}}^{\mathcal{J}}$
bottom	$\bot$	$\emptyset$
concept name	CN	$\mathsf{CN}^{\mathcal{J}} \subseteq \Delta_{A_{\mathrm{i}-1}}^{\mathcal{J}}$
general negation	$\neg_{\mathrm{i}} C$	$\Delta_{A_{\mathrm{i}-1}}^{\mathcal{J}} \setminus C^{\mathcal{J}}$
conjunction	$C \sqcap_{\mathrm{i}} D$	$C^{\mathcal{J}} \cap D^{\mathcal{J}}$
disjunction	$C \sqcup_{\mathrm{i}} D$	$C^{\mathcal{J}} \cup D^{\mathcal{J}}$
nominals	$\{\mathsf{o}\}$	$\{\mathsf{o}\}^{\mathcal{J}} = \{\mathsf{o}^{\mathcal{J}}\}$
exists restriction	$\exists_{\mathrm{i}} R.C$	$\{x \in \Delta_{A_{\mathrm{i}-1}}^{\mathcal{J}} \mid \exists y.\langle x, y\rangle \in R^{\mathcal{J}} \wedge y \in C^{\mathcal{J}}\}$
value restriction	$\forall_{\mathrm{i}} R.C$	$\{x \in \Delta_{A_{\mathrm{i}-1}}^{\mathcal{J}} \mid \forall y.\langle x, y\rangle \in R^{\mathcal{J}} \rightarrow y \in C^{\mathcal{J}}\}$
atleast restriction	$\geqslant_{\mathrm{i}} mR$	$\{x \in \Delta_{A_{\mathrm{i}-1}}^{\mathcal{J}} \mid \sharp\{y \mid \langle x, y\rangle \in R^{\mathcal{J}}\} \geq m\}$
atmost restriction	$\leqslant_{\mathrm{i}} mR$	$\{x \in \Delta_{A_{\mathrm{i}-1}}^{\mathcal{J}} \mid \sharp\{y \mid \langle x, y\rangle \in R^{\mathcal{J}}\} \leq m\}$
datatype exists restriction	$\exists_1 T.d$	$\{x \in \Delta_{A_0}^{\mathcal{J}} \mid \exists t.\langle x, t\rangle \in T^{\mathcal{J}} \wedge t \in d^{\mathcal{J}}\}$
datatype value restriction	$\forall_1 T.d$	$\{x \in \Delta_{A_0}^{\mathcal{J}} \mid \forall t.\langle x, t\rangle \in T^{\mathcal{J}} \rightarrow t \in d^{\mathcal{J}}\}$
datatype atleast restriction	$\geqslant_1 mT$	$\{x \in \Delta_{A_0}^{\mathcal{J}} \mid \sharp\{t \mid \langle x, t\rangle \in T^{\mathcal{J}}\} \geq m\}$
datatype atmost restriction	$\leqslant_1 mT$	$\{x \in \Delta_{A_0}^{\mathcal{J}} \mid \sharp\{t \mid \langle x, t\rangle \in T^{\mathcal{J}}\} \leq m\}$

Table 1. OWL FA classes

The semantics of OWL FA-classes are presented in Table 1 (page 3). C is *satisfiable* iff there exist an interpretation $\mathcal{J}$ s.t. $C^{\mathcal{J}} \neq \emptyset$; C subsumes D iff for every interpretation $\mathcal{J}$ we have $C^{\mathcal{J}} \subseteq D^{\mathcal{J}}$.

An OWL FA knowledge base Σ consists of $\Sigma_1, \ldots, \Sigma_k$. Each Σ_{i} consists of a TBox $\mathcal{T}_{\mathrm{i}}$, an RBox $\mathcal{R}_{\mathrm{i}}$ and an ABox $\mathcal{A}_{\mathrm{i}}$. Due to space limitation, we only provide details of OWL FA ABox here; it is obvious to extend traditional class and property axioms to meta-class axioms and meta-property axioms by introducing stratum numbers. Let $\mathsf{a}, \mathsf{b} \in \mathbf{V}_{\mathrm{I}}$ be individuals, C_1 a class in stratum 1, R_1 an abstract property in stratum 1, l a literal, $T \in \mathbf{V}_{\mathrm{D}}$ a datatype property, X, Y classes or abstract properties in stratum i, E a class in stratum i + 1 and S an abstract property in stratum i+1. An OWL FA *ABox* $\mathcal{A}_1$ is a finite set of individual axioms of the following forms: $\mathsf{a} :_1 C_1$, called *class assertions*, $\langle \mathsf{a}, \mathsf{b}\rangle :_1 R_1$, called *abstract property assertions*, $\langle \mathsf{a}, l\rangle :_1 T$, called *datatype property assertions*, $\mathsf{a} = \mathsf{b}$, called *individual equality axioms* and , $\mathsf{a} \neq \mathsf{b}$, called *individual inequality axioms*. An interpretation $\mathcal{J}$ satisfies $\mathsf{a} :_1 C_1$ if $\mathsf{a}^{\mathcal{J}} \in C_1^{\mathcal{J}}$; it satisfies $\langle \mathsf{a}, \mathsf{b}\rangle :_1 R_1$ if $\langle \mathsf{a}^{\mathcal{J}}, \mathsf{b}^{\mathcal{J}}\rangle \in R_1^{\mathcal{J}}$; it satisfies $\langle \mathsf{a}, l\rangle :_1 T$ if $\langle \mathsf{a}^{\mathcal{J}}, l^{\mathcal{J}}\rangle \in T^{\mathcal{J}}$; it satisfies $\mathsf{a} = \mathsf{b}$ if $\mathsf{a}^{\mathcal{J}} = \mathsf{b}^{\mathcal{J}}$; it satisfies $\mathsf{a} \neq \mathsf{b}$ if $\mathsf{a}^{\mathcal{J}} \neq \mathsf{b}^{\mathcal{J}}$. An OWL FA *ABox* $\mathcal{A}_{\mathrm{i}}$ is a finite set of axioms of the following forms: $X : E$, called *meta-class assertions*, $\langle X, Y\rangle : R$, called *meta-property assertions*, or $X =_{\mathrm{i}-1} Y$, called *meta individual equality axioms*. An interpretation $\mathcal{J}$ satisfies $X : E$ if $X^{\mathcal{J}} \in E^{\mathcal{J}}$; it satisfies $\langle X, Y\rangle : R$ if $\langle X^{\mathcal{J}}, Y^{\mathcal{J}}\rangle \in R^{\mathcal{J}}$; it satisfies $X =_{\mathrm{i}-1} Y$ if $X^{\mathcal{J}} = Y^{\mathcal{J}}$. Note that there are no meta-individual inequality axioms in OWL FA; it is more intuitive for users to apply disjoint class axioms.

According to the above definition, it is obvious that Σ_1 is a $\mathcal{SHOIN}(\mathbf{D})$ knowledge base, and Σ_2, …, Σ_{k} are $\mathcal{SHIQ}$ knowledge bases. Note that classes and property names in Σ_{i} are treated as individual names in $\Sigma_{\mathrm{i}+1}$; therefore, class and property equality axioms in Σ_{i} can act as individual equality axioms in $\Sigma_{\mathrm{i}+1}$. On the other hand, individual equalities explicitly asserted and implicitly entailed by number restrictions in $\Sigma_{\mathrm{i}+1}$ can act as class and property equality axioms in Σ_{i}.

Definition 1. *Let $\Sigma = \langle \Sigma_1, \ldots, \Sigma_{\mathrm{k}}\rangle$ be an OWL FA knowledge base, where each of $\Sigma_1, \ldots, \Sigma_{\mathrm{k}}$ is consistent. $\Sigma^* = \langle \Sigma_1^*, \ldots, \Sigma_{\mathrm{k}}^*\rangle$, called the* explicit knowledge base, *is constructed by making all the implicit atomic class axioms, atomic property axioms and individual equality axioms explicit.* ⋄

[3] To simplify the presentation, we do not distinguish datatype names and datatype URIrefs here.

As we have a finite set of vocabulary, we have the following Lemma.

Lemma 1. *Given an OWL FA knowledge base $\Sigma = \langle \Sigma_1, \ldots, \Sigma_{\mathrm{k}} \rangle$. Σ^* can be constructed from Σ in finite steps.*

Proof (sketch)**:** When $\mathrm{k} = 1$, it is easy to show that we can calculate the explicit knowledge base Σ_1^* in finite steps because the sets of names of classes (in stratum 1), roles (in stratum 1) and individuals are finite. When $\mathrm{k} > 1$, let us assume that we can calculate the explicit knowledge bases $\Sigma_1', ..., \Sigma_{\mathrm{i}}'$ (where $1 \leq \mathrm{i} < \mathrm{k}$) from $\Sigma_1, \ldots, \Sigma_{\mathrm{i}}$ in finite steps. We add all the class and property equality axioms in Σ_{i}' to $\Sigma_{\mathrm{i}+1}$. If the updated $\Sigma_{\mathrm{i}+1}$ is consistent, we can make the implicit individual equality axioms (if any) explicit and add new class and property equality axioms into Σ_{i}'. According to our assumption, we can calculate $\Sigma_1'', ..., \Sigma_{\mathrm{i}}''$ in finite steps. As the individual names in $\Sigma_{\mathrm{i}+1}$ are finite, we can calculate the explicit knowledge bases $\Sigma_1^*, \ldots, \Sigma_{\mathrm{i}+1}^*$ in finite steps.

Theorem 1. *Given an OWL FA knowledge base $\Sigma = \langle \Sigma_1, \ldots, \Sigma_{\mathrm{k}} \rangle$ and a class C in stratum* i*, C is satisfiable w.r.t. Σ iff C is satisfiable w.r.t. Σ_{i}^*.*

Theorem 2. *Given an OWL FA knowledge base $\Sigma = \langle \Sigma_1, \ldots, \Sigma_{\mathrm{k}} \rangle$. Σ is satisfiable iff each Σ_{i}^* ($1 \leq i \leq k$) is satisfiable.*

Theorem 2 indicates we can reduce the OWL FA-knowledge base satisfiability problem to the OWL DL-knowledge base satisfiability problem.

3 Related Work

[3] also provides two alternative metamodeling approaches for OWL DL, i.e., the context approach and the HiLog approach. In the context approach, the names for classes, properties and individuals are not distinct and are interpreted depending on the context; i.e., they are interpreted by class interpretation functions, property interpretation functions and individual interpretation functions, respectively. The HiLog approach is closer to the spirit of OWL Full metamodeling. Datatypes are not covered in these two approaches. We now use an example in [3] to illustrate some of the differences among the above two approaches and our approach. Let us consider the following knowledge base[4] $\Sigma = \{\, \mathtt{Harry} :_1 \mathsf{Eagle}, \mathtt{Harry} :_1 \neg\mathsf{Aquila}, \mathsf{Eagle} =_1 \mathsf{Aquila} \}$. In the context approach, since Eagle and Aquila as concepts and as individuals are independent, Σ is satisfiable. In the HiLog approach, it is not satisfiable because Eagle and Aquila are interpreted as the same object, let us call it a, and $\mathtt{Harry}$ cannot be both in and not in the concept extension of a. In OWL FA, Σ is unsatisfiable because the meta-individual equality axiom $\mathsf{Eagle} =_1 \mathsf{Aquila}$ indicates two concepts Eagle and Aquila are equivalent, and $\mathtt{Harry}^{\mathcal{J}}$ cannot be both in and not in $\mathsf{Eagle}^{\mathcal{J}}$.

4 Conclusion and Outlook

In this paper, we propose the OWL FA ontology language as a metamodeling extension of OWL DL, using the metamodeling architecture of RDFS(FA), which is very similar to that of UML. The syntax of OWL FA is very similar to that of OWL DL; the former introduces a stratum number to attach to OWL FA class constructors and axioms. The semantics of OWL FA is a natural extension of that of OWL DL, dividing the abstract domain into k sub-domains for k strata. We have shown that OWL FA is decidable, and its basic reasoning tasks can be reduced to that of OWL DL. In the future, we plan to implement the construction of extended knowledge bases so that we can use OWL DL reasoners to reason with OWL FA ontologies, and to evaluate it with, for example, the WordNet ontology.

[4] In [3], the subscripts are not used.

Bibliography

[1] Patrick Hayes. RDF Semantics. Technical report, W3C, Feb 2004. W3C recommendation, http://www.w3.org/TR/rdf-mt/.

[2] Ian Horrocks, Peter F. Patel-Schneider, and Frank van Harmelen. From SHIQ and RDF to OWL: The making of a web ontology language. *Journal of Web Semantics*, 1(1):7–26, 2003.

[3] Boris Motik. On the Properties of Metamodeling in OWL. In *Proc. of the Fourth International Semantic Web Conference (ISWC2005)*, 2005.

[4] Jeff Z. Pan. *Description Logics: Reasoning Support for the Semantic Web*. PhD thesis, School of Computer Science, The University of Manchester, 2004.

[5] Jeff Z. Pan and Ian Horrocks. RDFS(FA) and RDF MT: Two Semantics for RDFS. In *Proc. of the 2nd International Semantic Web Conference (ISWC2003)*, 2003.

[6] Peter F. Patel-Schneider. Layering the Semantic Web: Problems and Directions. In *2002 International Semantic Web Conference*, Jun 2002.

[7] Peter F. Patel-Schneider. Two Proposals for a Semantic Web Ontology Language. In *2002 International Description Logic Workshop*, Apr 2002.

[8] Peter F. Patel-Schneider, Patrick Hayes, and Ian Horrocks. OWL Web Ontology Language Semantics and Abstract Syntax. Technical report, W3C working draft, Mar. 2003.

[9] Guus Schreiber. The Web is not well-formed. *IEEE Intelligent Systems*, 17(2):79–80, 2002. Contribution to the section "Trends and Controversies: Ontologies KISSES in Standardization".

[10] Jan Wielemaker, Guus Schreiber, and Bob Wielinga. Prolog-based infrastructure for RDF: performance and scalability. In *Proc. of the 2nd International Semantic Web Conference (ISWC2003)*, pages 644–658, 2003.