Metadata-driven selection and integration of object-level knowledge

Paul Groth, Szymon Klarman, Stefan Schlobach, and Jacco van Ossenbruggen

Department of Computer Science
Vrije Universiteit Amsterdam, The Netherlands
{p.t.groth, s.klarman, k.s.schlobach, j.r.van.ossenbruggen}@vu.nl

Abstract. Structured information about an application domain (*object-level knowledge*) can be represented on the Semantic Web in the form of an OWL/RDFS ontology and made accessible for applications to query and reason over. In practice, the abundance of heterogenous knowledge sources addressing very similar or overlapping domains makes it hard to identify the relevant pieces of information which should be selected and integrated in order to answer a particular query in the context of a given application. One natural way of alleviating this problem is to utilize the metadata (*meta-level knowledge*), i.e. formal description of object-level representations regarding their content, provenance and context of applicability. We propose a simple, generic framework, based on standard formalisms, supporting an interacting representation of both levels of knowledge and propose a novel form of queries which are executed in two well-defined steps: (meta-level) ontology selection and (object-level) query answering. The framework is expressed purely as a composition of standard Semantic Web languages, and the complexity of reasoning in the framework does not exceed that of reasoning in the underlying languages. To demonstrate the ease of use and simplicity of our formal approach we report on an implementation as a Large Knowledge Collider workflow. Finally, the approach is motivated with a real-life use case.

1 Introduction

As the adoption of Semantic Web approaches has grown so has the availability of large amounts of overlapping knowledge sources pertaining to the same domain. For example, in the Web of Data, we see macro clusters of knowledge in diverse areas from government and research to music and biomedicine. At a micro level, we see ontologies being progressively updated and multiple versions of the same ontology being used simultaneously. Additionally, knowledge is often being generated by a variety of different mechanisms from automated mapping techniques to expert entry. For instance, the sig.ma search engine [1], at the time of writing, returns twenty different knowledge sources used to describe the concept of "heart disease", ranging from Wikipedia to Examiner.com, a local news site, to slide sets from anonymous users and the Pew Internet Trust.

In this environment, applications developers are faced with a challenge, how does one select and integrate the right set of *object-level knowledge* while not statically encoding which knowledge to use. Applications for example may want to focus on up-to-date knowledge, knowledge generated by particular software mechanisms, or knowledge provided by a particular organization. This *meta-knowledge* is key to being able to select the right set of knowledge to be used within the application. In practice, applications often encode the decisions about which object-level knowledge to use either in an off-line selection process or in every query they issue to an integrated knowledge base. Thus, developers are faced with either less flexible approaches or increased query complexity. Furthermore, these approaches provide no formal grounding about the consequences of reasoning when integrating knowledge. Specifically, we formulate the problem as follows: *How does one systematically, rigorously and simply deploy meta-knowledge in order to facilitate selective reasoning over object-level knowledge?*

To address this problem, we present a framework for the selection and integration of object-level knowledge based on formally modeled meta-knowledge. The framework provides three crucial benefits:

1. it has a clear formal grounding ensuring guarantees that reasoning complexity does not exceed that of the underlying languages used;
2. it builds upon widely deployed Semantic Web representations and tools;
3. it is timely, as many methodologies for building semantic datasets come with formal annotations such as OPMV and VOiD, which are ready for use in the framework.

Our framework thus strikes a balance between theoretical rigor and ease of implementation. To emphasize this ease of use and implementation we have built our framework using an existing Semantic Web development platform, the Large Knowledge Collider [2], and explain its potential with a use case study from the automated alignment of the Wordnet vocabulary for the cultural heritage domain.

In summary, the contributions of this paper are as follows:

1. A generic formal framework combining two key features: representation and reasoning with meta-knowledge and integration of multiple, context-specific object knowledge representations.
2. The first such framework expressed purely in terms of compositions of standard SW representations (DL/OWL/RDFS ontologies).
3. Formal results showing that the framework's reasoning complexity does not exceed the underlying languages.
4. An implementation of the framework that shows that the framework can be easily deployed using an existing Semantic Web development platform.

The rest of the paper is organized as follows, we begin by presenting the framework and its formal properties. An implementation of the framework is then described. We follow that with a description of the case study and the application of the framework to it. We then address related work particularly emphasizing other formal approaches. Finally, we discuss future work and conclude.

2 The SIS$^{/\mathrm{MD}}$ Framework

In this section we define the framework and the novel type of queries associated with it. We start with a high-level overview of the adopted formalization and then elaborate its technical aspects, including the syntax, semantics and basic computational properties. Further, we introduce the querying mechanism.

2.1 Overview

The framework supports integration of multiple representation systems containing possibly fragmentary and heterogenous object-level knowledge, with a parallel representation of the meta-level knowledge over those systems, regarding their content, provenance and/or contextual information. Reasoning over the framework intertwines, in a controlled manner, inference over these two levels. Importantly, the framework is reducible to existing formalisms and reasoning problems, which ensures strong and well-understood formal foundations and relatively straightforward implementations.

The formulation of our approach is sufficiently generic to permit most current Semantic Web languages for modelling object and meta-ontologies. This includes all OWL dialects [3,4] that are underpinned by model-theoretic semantics and the atomic terms of which include concepts/classes, roles/properties and individuals/instances. To keep the presentation uniform, unless explicitly stated otherwise, we refer to Description Logics (DLs) [5] as the assumed representation language for both levels of knowledge involved.

The knowledge models supported by the presented framework shall be denoted as *Simple Interoperability Systems with Meta-Data* (SIS$^{/\mathrm{MD}}$). The central components of a SIS$^{/\mathrm{MD}}$, as illustrated in Figure 1, are:

***object ontologies*:** formal representations of different portions of object-level knowledge about an application domain,
***meta-ontology*:** formal representation of meta-level knowledge about the object ontologies.

The object ontologies are standard DL ontologies which can be metaphorically depicted as "boxes" [6]. Each box is equipped with its own vocabulary and associated with a unique formal entity called a *context.* A box contains a portion of domain knowledge specific to its context. Boxes can be integrated by sharing their local vocabularies. A shared term must be given the same semantic interpretation in all the boxes in which it appears. This approach is motivated by the typical solutions found in the Web environment, where contexts correspond to the URIs of ontologies and allow one to import pieces of vocabulary from different sources. The meta-ontology is another DL ontology, in which the contexts are represented as individuals. A box is thus given a two-fold representation in a SIS$^{/\mathrm{MD}}$: on the meta-level it is treated as an atomic individual described in the meta-language — on the object level it is associated with a single ontology.

The semantics of the framework is grounded directly in the standard model-theoretic semantics of the languages used on the object and the meta-level of

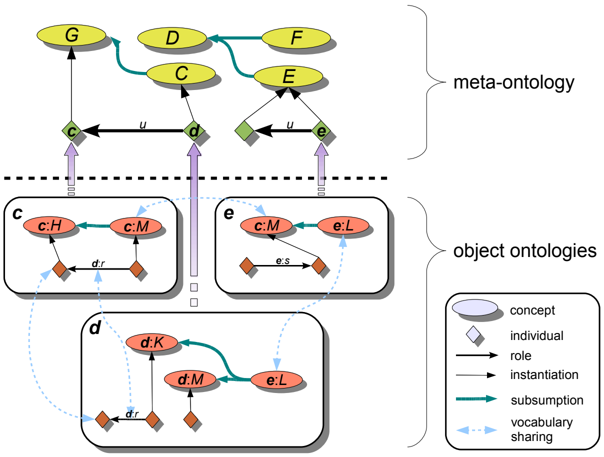

Fig. 1: A Simple Interoperability System with Meta-Data.

representation. Rather than extending or redesigning these semantic foundations, we follow a simple, compositional approach. A model of a SIS$^{/\mathrm{MD}}$ is a *composition of* (standard) *models* of the ontologies included in the SIS$^{/\mathrm{MD}}$, which must satisfy certain compatibility criteria. The formal characteristics of the framework are determined largely by the following two properties.

1. The semantic relationship between the object and the meta-level of the representation is *largely conventional*, i.e. it involves no genuine formal interaction between the semantics of both levels.
2. The semantic interoperability mechanism used for relating the contents of the object ontologies is of a purely *extensional character*, in the sense that two ontologies can be semantically related only by interpreting some parts of their vocabularies via identical extensions in a (global) interpretation domain.

Figure 2 presents a sample model of (a part of) the SIS$^{/\mathrm{MD}}$ used in Figure 1. To witness the first of the properties above, observe that the model of the meta-ontology is strictly disconnected from the models of the object ontologies. The only relationship between the two levels is that some of the objects appearing in the model of the meta-ontology *are conventionally associated* with the corresponding object ontologies. Formally, however, the problem of satisfiability of the meta-ontology, i.e. verifying existence of its model, is fully independent from the satisfiability of the object ontologies. This separation guarantees good computational properties of the framework, namely reducibility of reasoning to standard decision problems in the underlying languages and, consequently, preservation of the complexity of reasoning. At the same time even such loose composition provides enough expressive power to support an interesting form of querying, in which results of reasoning on the meta-level can be further used for specializing the reasoning tasks on the object level.

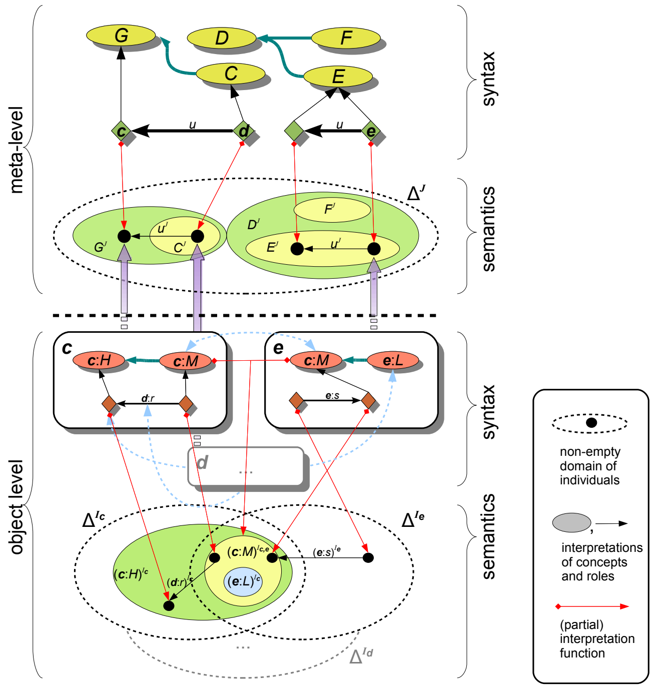

Fig. 2: The semantics of a SIS$^{/\mathrm{MD}}$.

The second property is illustrated in the figure with a sample use of concept $\boldsymbol{c}{:}M$, which occurs simultaneously in the boxes labeled with $\boldsymbol{c}$ and $\boldsymbol{e}$. In such case, the corresponding models for the two ontologies need to agree on the interpretation of $\boldsymbol{c}{:}M$, where the agreement is understood as having identical extensions. More precisely, whenever α is a common term for some object ontologies $\mathcal{O}_{\boldsymbol{c}}$ and $\mathcal{O}_{\boldsymbol{e}}$ in some SIS$^{/\mathrm{MD}}$, then their models $\mathcal{I}_{\boldsymbol{c}} = (\Delta^{\mathcal{I}_{\boldsymbol{c}}}, \cdot^{\mathcal{I}_{\boldsymbol{c}}})$ and $\mathcal{I}_{\boldsymbol{e}} = (\Delta^{\mathcal{I}_{\boldsymbol{e}}}, \cdot^{\mathcal{I}_{\boldsymbol{e}}})$ can be incorporated into a global model of the SIS$^{/\mathrm{MD}}$ only if it is the case that $\alpha^{\mathcal{I}_{\boldsymbol{c}}} = \alpha^{\mathcal{I}_{\boldsymbol{e}}}$. Note, that we do not require the domains of the models to be identical as well, but only to overlap in the fragments in which the shared vocabulary is interpreted. The obvious intuition is that it must be possible to coherently merge the local models of all the object ontologies into a single global model of the whole object-level knowledge represented in a SIS$^{/\mathrm{MD}}$.

2.2 Syntax and semantics

We start be recapping the basic nomenclature of DLs. A DL language $\mathcal{L}$ is defined by its vocabulary $\Sigma = (N_C, N_R, N_I)$, where N_C is a set of concept names, N_R a set of role names and N_I a set of individual names, and a selection of logical operators enabling construction of complex concepts, roles and axioms.

Different combinations of operators give rise to DLs of different expressiveness, and consequently, of different computational complexity, from the highly expressive $\mathcal{SROIQ}$, underpinning the OWL 2 DL language, to the lightweight $\mathcal{EL}^{++}$ or the DL-*Lite* family, on which restricted OWL profiles are based [4]. Table 1 presents a sample of concept constructors and axioms available in DLs and their rendering into the OWL/RDF(S) syntax.

The model-theoretic semantics of $\mathcal{L}$ is given through interpretations of the form $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a non-empty domain of individuals and $\cdot^{\mathcal{I}}$ is an interpretation function which maps $C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$, for every $C \in N_C$, $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, for every $r \in N_R$ and $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, for every $a \in N_I$. Further, it is inductively extended over complex expressions. A DL ontology $\mathcal{O}$ in the language $\mathcal{L}$ is a set of axioms in $\mathcal{L}$. An axiom is satisfied by an interpretation *iff* the semantic condition associated with the axiom holds in that interpretation (see Table 1 for examples). We say that an interpretation is a model of an ontology *iff* it satisfies all axioms in this ontology.

Syntax	Semantics	OWL/RDF(S) constructor
Concept/Class constructors		
$\top$	$\Delta^{\mathcal{I}}$	owl:Thing
$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$	owl:complementOf
$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$	owl:intersectionOf
$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$	owl:unionOf
Axioms / Interpretation constraints		
$C(a)$	$a^{\mathcal{I}} \in C^{\mathcal{I}}$	rdf:type
$r(a, b)$	$(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$	*RDF triple syntax*
$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$	rdfs:subClassOf
$C \equiv D$	$C^{\mathcal{I}} = D^{\mathcal{I}}$	owl:equivalentClass
$r \sqsubseteq s$	$r^{\mathcal{I}} \subseteq s^{\mathcal{I}}$	rdfs:subPropertyOf
$a = b$	$a^{\mathcal{I}} = b^{\mathcal{J}}$	owl:sameAs
$\mathrm{dom}(r) \sqsubseteq C$	$\{x \in \Delta^{\mathcal{I}} \mid (x, y) \in r^{\mathcal{I}}\} \subseteq C^{\mathcal{I}}$	rdfs:domain
$\mathrm{ran}(r) \sqsubseteq C$	$\{y \in \Delta^{\mathcal{I}} \mid (x, y) \in r^{\mathcal{I}}\} \subseteq C^{\mathcal{I}}$	rdfs:range

Table 1: Syntax and semantics of DL concepts and axioms.

We now formally introduce components of the SIS$^{/\mathrm{MD}}$ framework. A metalanguage is a DL with the standard syntax and semantics, distinguishable only by the function it plays in the framework. From this perspective, some individual names in this language should be seen as names of context boxes containing object knowledge. To avoid ambiguities in the context naming we assume the metalanguage adheres to the Unique Name Assumption.

Definition 1 (Metalanguage and meta-ontology). *A* metalanguage $\mathcal{L}_{\mathcal{M}}$ *is a DL language over vocabulary* $\Gamma = (M_C, M_R, M_I)$. *A* meta-ontology $\mathcal{M}$ *is a set of axioms in* $\mathcal{L}_{\mathcal{M}}$.

The definition of the object language introduces additional context prefixes for distinguishing between box-specific vocabularies, which make the interoperability mechanism more explicit in the formalization.

Definition 2 (Object language and ontologies). *Let* $\mathcal{L}_{\mathcal{M}}$ *be a meta-language over* $\Gamma = (M_C, M_R, M_I)$. *Then an* object language $\mathcal{L}_{\mathcal{O}}$ *under* $\mathcal{L}_{\mathcal{M}}$ *is a DL language over the vocabulary* $\Sigma^{\mathcal{L}_{\mathcal{M}}} = (N_C^{\mathcal{L}_{\mathcal{M}}}, N_R^{\mathcal{L}_{\mathcal{M}}}, N_I^{\mathcal{L}_{\mathcal{M}}})$, *such that* $\Sigma^{\mathcal{L}_{\mathcal{M}}}$ *is defined in terms of* M_I *and some DL vocabulary* $\Sigma = (N_C, N_R, N_I)$ *as follows:*

- $N_C^{\mathcal{L}_{\mathcal{M}}} = \{\boldsymbol{c}{:}C \mid \boldsymbol{c} \in M_I,\ C \in N_C\}$, $\quad N_R^{\mathcal{L}_{\mathcal{M}}} = \{\boldsymbol{c}{:}r \mid \boldsymbol{c} \in M_I,\ r \in N_R\}$,
- $N_I^{\mathcal{L}_{\mathcal{M}}} = \{\boldsymbol{c}{:}a \mid \boldsymbol{c} \in M_I,\ a \in N_I\}$.

The elements of the resulting sets $N_C^{\mathcal{L}_{\mathcal{M}}}$, $N_R^{\mathcal{L}_{\mathcal{M}}}$, $N_I^{\mathcal{L}_{\mathcal{M}}}$ *are concept names, role names and individual names, respectively. An* object ontology $\mathcal{O}$ *is a set of axioms in* $\mathcal{L}_{\mathcal{O}}$.

$\mathcal{I} = \{\mathcal{I}_{\boldsymbol{c}}\}_{\boldsymbol{c} \in M_I}$ *is an interpretation of* $\mathcal{L}_{\mathcal{O}}$ iff *for every* $\boldsymbol{c} \in M_I$ *it holds that:*

- $\mathcal{I}_{\boldsymbol{c}} = (\Delta^{\mathcal{I}_c}, \cdot^{\mathcal{I}_c})$ *is an interpretation of* Σ,
- *for every* $\boldsymbol{d}{:}\alpha \in \Sigma^{\mathcal{L}_{\mathcal{M}}}$, $(\boldsymbol{d}{:}\alpha)^{\mathcal{I}_c} = \alpha^{\mathcal{I}_d}$,
- $\cdot^{\mathcal{I}_c}$ *is inductively extended over all complex expressions of* $\mathcal{L}_{\mathcal{O}}$ *in the usual manner.*

The satisfaction relation and the notion of model are defined as usual. The design of an object language aims at capturing precisely the following intuition: a vocabulary Σ, interpreted over the object domain, might be used differently in different contexts. To avoid ambiguities, instead of referring to a plain atom $\alpha \in \Sigma$, one should rather use it in combination with a prefix $\boldsymbol{c} \in M_I$, explicitly indicating the intended context of interpretation. Effectively, the vocabulary of the object language can be restated as the set of all prefixed atoms $\Sigma^{\mathcal{L}_{\mathcal{M}}} = \{\boldsymbol{c}{:}\alpha \mid \boldsymbol{c} \in M_I,\ \alpha \in \Sigma\}$, where atoms with the same prefix $\boldsymbol{c} \in M_I$ are interpreted by a unique, designated DL interpretation $\mathcal{I}_{\boldsymbol{c}} = (\Delta^{\mathcal{I}_c}, \cdot^{\mathcal{I}_c})$. All complex expressions, comprising atoms with possibly different prefixes, are given their meaning simply by combining the respective interpretations.

Finally, we define the target notion of SIS$^{/\mathrm{MD}}$.

Definition 3 (SIS$^{/\mathrm{MD}}$)**.** *Let* $\mathcal{L}_{\mathcal{M}}$ *be a metalanguage and* $\mathcal{L}_{\mathcal{O}}$ *an object language under* $\mathcal{L}_{\mathcal{M}}$. *Then a tuple* $\mathcal{S} = \langle \mathcal{M}, M_I^\star, \{\mathcal{O}_{\boldsymbol{c}}\}_{\boldsymbol{c} \in M_I^\star} \rangle$ *is a* Simple Interoperability System with Meta-Data, *where* $\mathcal{M}$ *is a meta-ontology in* $\mathcal{L}_{\mathcal{M}}$, $M_I^\star \subseteq M_I$ *and for every* $\boldsymbol{c} \in M_I^\star$, $\mathcal{O}_{\boldsymbol{c}}$ *is an object ontology in* $\mathcal{L}_{\mathcal{O}}$.

Note that each object ontology $\mathcal{O}_{\boldsymbol{c}}$ in a SIS$^{/\mathrm{MD}}$ is uniquely identifiable by the corresponding context name $\boldsymbol{c} \in M_I$ and represents the object knowledge *directly* associated with that context. It is important to observe, however, that due to the employed interoperability mechanism this knowledge can be also carried *indirectly* by axioms included in other ontologies. For instance, axiom $\boldsymbol{c}{:}C \sqsubseteq \boldsymbol{c}{:}D \in \mathcal{O}_{\boldsymbol{d}}$, although included in the ontology $\mathcal{O}_{\boldsymbol{d}}$, imposes a constraint on the interpretation of the context $\boldsymbol{c}$, as it uses the vocabulary of $\boldsymbol{c}$. Thus, the names from M_I play a twofold role in the framework. On the semantic side, as denotations of the prefixes used in the object vocabulary, they determine the *logical space* of contexts relevant for interpreting the object knowledge. On the syntactic level, as identifiers for object ontologies, they also allow for enumerating *portions of data* which actually convey that knowledge.

The semantics of a SIS$^{/\mathrm{MD}}$ is defined straightforwardly by combining the semantics of both levels of representation.

Definition 4 (Semantics). *A pair $\langle \mathcal{J}, \mathcal{I} \rangle$ is an interpretation of a* SIS$^{/\mathrm{MD}}$ $\mathcal{S} = \langle \mathcal{M}, M_I^\star, \{\mathcal{O}_{\boldsymbol{c}}\}_{c \in M_I^\star} \rangle$ iff $\mathcal{J} = (\Delta^{\mathcal{J}}, \cdot^{\mathcal{J}})$ *is an interpretation of the metalanguage and $\mathcal{I} = \{\mathcal{I}_{\boldsymbol{c}}\}_{c \in M_I}$ is an interpretation of the object language of $\mathcal{S}$. It is a model of $\mathcal{S}$* iff *$\mathcal{J}$ satisfies all axioms in $\mathcal{M}$ and for every $\boldsymbol{c} \in M_I^\star$, $\mathcal{I}_{\boldsymbol{c}}$ satisfies all axioms in $\mathcal{O}_{\boldsymbol{c}}$.*

Example. Consider a system $\mathcal{S} = \langle \mathcal{M}, M_I^\star, \{\mathcal{O}_{\boldsymbol{c}}\}_{c \in M_I^\star} \rangle$, which integrates information about hospital patients with some biomedical knowledge ontologies. We set $M_I^\star = \{\boldsymbol{c}, \boldsymbol{d}, \boldsymbol{e}, \boldsymbol{f}\}$ and define $\mathcal{M}$ and $\{\mathcal{O}_{\boldsymbol{c}}\}_{c \in M_I^\star}$ as follows:

$\mathcal{M}$:	$\boldsymbol{MedicalOntology} \sqsubseteq \boldsymbol{BiomedicalKnowledge}$ $\boldsymbol{AnatomyOntology} \sqsubseteq \boldsymbol{BiomedicalKnowledge}$ $\boldsymbol{PatientKB}(\boldsymbol{c})$ $\boldsymbol{MedicalOntology}(\boldsymbol{d}),\ \ \boldsymbol{author}(\boldsymbol{d}, \boldsymbol{ihtsd\_organization})$ $\boldsymbol{AnatomyOntology}(\boldsymbol{e}),\ \ \boldsymbol{date}(\boldsymbol{e}, \boldsymbol{january2010})$ $\boldsymbol{Mappings}(\boldsymbol{f}),\ \exists \boldsymbol{generatedBy}.\boldsymbol{ManualMethod}(\boldsymbol{f})$
$\mathcal{O}_c$:	$\boldsymbol{c}{:}\mathit{CardiacPatient} \equiv \boldsymbol{c}{:}\mathit{Patient} \sqcap \exists \boldsymbol{c}{:}\mathit{diagnosedWith}.\boldsymbol{d}{:}\mathit{HeartDisease}$ $\boldsymbol{c}{:}\mathit{CardiacPatient}(\boldsymbol{c}{:}\mathit{johnSmith})$
$\mathcal{O}_d$:	$\boldsymbol{d}{:}\mathit{HeartDisease} \sqsubseteq \exists \boldsymbol{d}{:}\mathit{disorderOf}.\boldsymbol{d}{:}\mathit{Heart}$
$\mathcal{O}_e$:	$\boldsymbol{e}{:}\mathit{HumanHeart} \sqsubseteq \boldsymbol{e}{:}\mathit{Heart}$
$\mathcal{O}_f$:	$\boldsymbol{d}{:}\mathit{Heart} \equiv \boldsymbol{e}{:}\mathit{HumanHeart}$ $\boldsymbol{c}{:}\mathit{johnSmith} = \boldsymbol{g}{:}\mathit{jSmith}$

The meta-ontology $\mathcal{M}$ above, represents the meta-knowledge over contexts integrated in the system. For instance, $\boldsymbol{e}$ is stated to be a context of anatomy, and thus falls in the scope of biomedical knowledge due to axiom $\boldsymbol{AnatomyOntology} \sqsubseteq \boldsymbol{BiomedicalKnowledge}$. Moreover, $\boldsymbol{e}$ is related to individual $\boldsymbol{january2010}$ via role $\boldsymbol{date}$. The object ontologies corresponding to the contexts contain fragments of object knowledge. For example, $\mathcal{O}_{\boldsymbol{d}}$ states that $\boldsymbol{d}{:}\mathit{HeartDisease} \sqsubseteq \exists \boldsymbol{d}{:}\mathit{disorderOf}.\boldsymbol{d}{:}\mathit{Heart}$, meaning that objects with heart disease have some disorder of heart, where all involved atoms are interpreted in the context $\boldsymbol{d}$. Atoms might also originate in external contexts, thus involving interoperability between different contexts, e.g. $\boldsymbol{d}{:}\mathit{HeartDisease}$ in $\mathcal{O}_{\boldsymbol{c}}$ or $\boldsymbol{e}{:}\mathit{HumanHeart}$ in $\mathcal{O}_{\boldsymbol{f}}$. In a special case of $\boldsymbol{g}{:}\mathit{jSmith}$ in $\mathcal{O}_{\boldsymbol{f}}$, we refer to the context $\boldsymbol{g}$ which is not associated with an ontology, but still belongs to the logical space of contexts. Note also, that ontology $\mathcal{O}_{\boldsymbol{f}}$ does not make use at all of its native vocabulary, but merely relates the vocabularies of other contexts. Such ontologies are essential for formalizing the notion of alignment in the framework. Sample inferences based on the system described above are shown in the following section.

2.3 Reasoning

Technically, reasoning over the framework boils down to reasoning over two separate representations. In particular, deciding satisfiability of a $\mathrm{SIS}^{/\mathrm{MD}}$ is equivalent to deciding two independent problems: satisfiability of the meta-ontology and satisfiability of the set of object ontologies. Both problems are solvable using existing reasoning tools, such as popular DL reasoners. In the former case, this is immediate, as the meta-ontology is a standard DL ontology. In the latter, the set of object ontologies in $\mathcal{L}_{\mathcal{O}}$ must be first reduced to a single, equisatisfiable DL ontology. Such polynomial reduction, which we outline in Table 2, is analogical to the one used for reducing Package-based DLs to DLs [7] (see Section 5) and applies directly to most DLs, including those underlying the OWL profiles. Note, that avoiding the reduction step and taking the union of the ontologies instead violates the local character of axioms and thus, in certain cases (mostly expressive DLs), might lead to unintended inferences or inconsistencies.

INPUT: Set of DL ontologies $\{\mathcal{O}_{\boldsymbol{c}}\}_{c\in M_I^\star}$ in $\mathcal{L}_{\mathcal{O}}$

1. For every $\boldsymbol{c} \in M_I^\star$, replace every occurrence of $\top$ in $\mathcal{O}_{\boldsymbol{c}}$ with a fresh concept $\boldsymbol{c}{:}\top$.
2. For every $\boldsymbol{c} \in M_I^\star$ and every occurring concept name $\boldsymbol{c}{:}C$, individual name $\boldsymbol{c}{:}a$ and role name $\boldsymbol{c}{:}r$, extend $\mathcal{O}_{\boldsymbol{c}}$ with the following axioms:
$$\boldsymbol{c}{:}C \sqsubseteq \boldsymbol{c}{:}\top, \qquad \boldsymbol{c}{:}\top(\boldsymbol{c}{:}a), \qquad \mathrm{dom}(\boldsymbol{c}{:}r) \sqsubseteq \boldsymbol{c}{:}\top, \qquad \mathrm{ran}(\boldsymbol{c}{:}r) \sqsubseteq \boldsymbol{c}{:}\top.$$
3. If $\mathcal{L}_{\mathcal{O}}$ supports the complement then for every $\boldsymbol{c} \in M_I^\star$:
   (a) replace every $C \equiv D \in \mathcal{O}_{\boldsymbol{c}}$ with $\boldsymbol{c}{:}\top \sqsubseteq \mathrm{NNF}((\neg C \sqcup D) \sqcap (C \sqcup \neg D))$, every $C \sqsubseteq D \in \mathcal{O}_{\boldsymbol{c}}$ with $\boldsymbol{c}{:}\top \sqsubseteq \mathrm{NNF}(\neg C \sqcup D)$, and every $C(a) \in \mathcal{O}_{\boldsymbol{c}}$ with $\mathrm{NNF}(C)(a)$, where NNF denotes the Negation Normal Form,
   (b) replace every occurrence of $\neg\boldsymbol{c}{:}C$ with a fresh concept $\widetilde{\boldsymbol{c}{:}C}$ and extend $\mathcal{O}_{\boldsymbol{c}}$ with the following axioms:
$$\widetilde{\boldsymbol{c}{:}C} \sqsubseteq \boldsymbol{c}{:}\top, \qquad \boldsymbol{c}{:}\top \sqsubseteq \widetilde{\boldsymbol{c}{:}C} \sqcup \boldsymbol{c}{:}C, \qquad \widetilde{\boldsymbol{c}{:}C} \sqcap \boldsymbol{c}{:}C \sqsubseteq \bot.$$
4. Set $\mathcal{O} = \bigcup_{c\in M_I^\star} \mathcal{O}_{\boldsymbol{c}}$.

OUTPUT: DL ontology $\mathcal{O}$

Table 2: Reduction of object ontologies in $\mathcal{L}_{\mathcal{O}}$ to an equisatisfiable DL ontology.

Similarly to the satisfiability problem, the notion of entailment can be defined independently for both levels as follows.

Definition 5 (Entailment). *A meta-ontology $\mathcal{M}$ entails a formula φ over the vocabulary of $\mathcal{L}_{\mathcal{M}}$, i.e. $\mathcal{M} \models \varphi$,* iff *every model $\mathcal{J} = (\Delta^{\mathcal{J}}, \cdot^{\mathcal{J}})$ of $\mathcal{M}$ satisfies φ.*

A set of object ontologies $\{\mathcal{O}_{\boldsymbol{c}}\}_{\boldsymbol{c}\in M_I^\star}$*, for* $M_I^\star \subseteq M_I$*, entails a formula* φ *over the vocabulary of* $\mathcal{L}_\mathcal{O}$*, i.e.* $\{\mathcal{O}_{\boldsymbol{c}}\}_{\boldsymbol{c}\in M_I^\star} \models \varphi$, iff *every model* $\mathcal{I} = \{\mathcal{I}_{\boldsymbol{c}}\}_{\boldsymbol{c}\in M_I}$ *of* $\{\mathcal{O}_{\boldsymbol{c}}\}_{\boldsymbol{c}\in M_I^\star}$ *satisfies* φ*.*

Regardless of the semantic separation of the two levels of representation, the framework supports a simple, yet practically useful form of queries which allow for metadata-driven selection, integration and reasoning over the object-level knowledge. SIS$^{/\mathrm{MD}}$ queries, of the form $m(y) : q(\overline{x})$, comprise a metalevel query $m(y)$ and an object-level query $q(\overline{x})$. The metalevel component serves for retrieving the object ontologies which satisfy certain meta-level descriptions. The object query is then applied over the fragment of knowledge contained in those ontologies. The schematic workflow for answering the queries in practical implementations is presented in detail in Figure 3, while formally the reasoning problem is defined as follows:

Definition 6 (SIS$^{/\mathrm{MD}}$ **query).** *An expression* $m(y) : q(\overline{x})$ *is a* SIS$^{/\mathrm{MD}}$ query *over an* SIS$^{/\mathrm{MD}}$ $= \langle \mathcal{M}, M_I^\star, \{\mathcal{O}_{\boldsymbol{c}}\}_{\boldsymbol{c}\in O}\rangle$ iff *the following conditions hold:*

- $m(y) \leftarrow \exists \overline{z}.\varphi(y, \overline{z})$ *is a query over the vocabulary of* $\mathcal{L}_\mathcal{M}$*,*
- $q(\overline{x}) \leftarrow \exists \overline{v}.\psi(\overline{x}, \overline{v})$ *is a query over the vocabulary of* $\mathcal{L}_\mathcal{O}$*.*

A sequence $\overline{a} \in N_I$ *is an* answer *to query* $m(y) : q(\overline{x})$ iff $\{\mathcal{O}_{\boldsymbol{c}}\}_{\boldsymbol{c}\in M} \models q(\overline{a})$*, where* $M = \{\boldsymbol{c} \in M_I^\star \mid \mathcal{M} \models m(\boldsymbol{c})\}$*.*

Here we use the standard query notation $q(\overline{x}) \leftarrow \exists \overline{y}.body(\overline{x}, \overline{y})$, where $\overline{x}$ denotes the free variables (answer variables) in a set of atoms *body*.

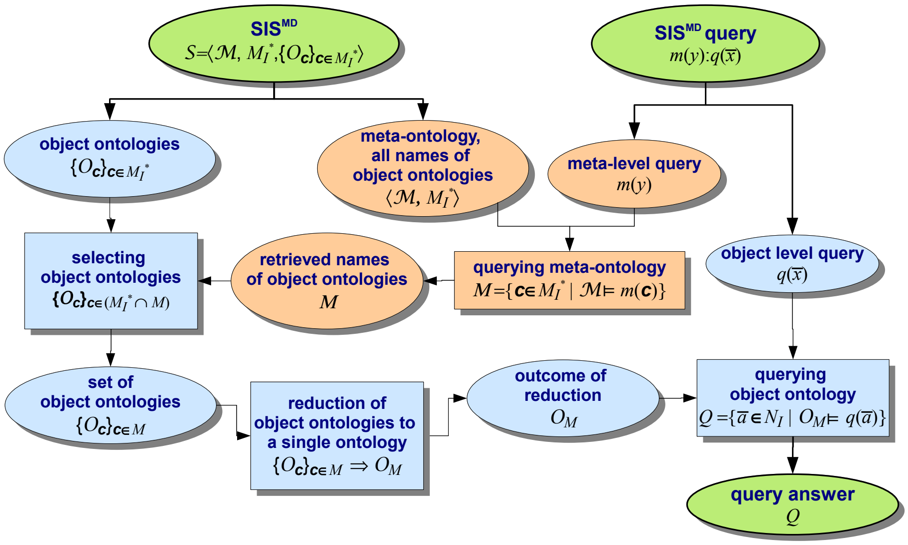
SIS^{MD}
$S=\langle \mathcal{M}, M_I^*, \{O_\mathbf{c}\}_{\mathbf{c}\in M_I^*}\rangle$
SIS^{MD} query
$m(y):q(\overline{x})$
object ontologies
$\{O_\mathbf{c}\}_{\mathbf{c}\in M_I^*}$
meta-ontology, all names of object ontologies
$\langle \mathcal{M}, M_I^*\rangle$
meta-level query
$m(y)$
selecting object ontologies
$\{O_\mathbf{c}\}_{\mathbf{c}\in (M_I^* \cap M)}$
retrieved names of object ontologies
M
querying meta-ontology
$M=\{\mathbf{c}\in M_I^* \mid \mathcal{M}\models m(\mathbf{c})\}$
object level query
$q(\overline{x})$
set of object ontologies
$\{O_\mathbf{c}\}_{\mathbf{c}\in M}$
reduction of object ontologies to a single ontology
$\{O_\mathbf{c}\}_{\mathbf{c}\in M} \Rightarrow O_M$
outcome of reduction
O_M
querying object ontology
$Q=\{\overline{a}\in N_I \mid O_M\models q(\overline{a})\}$
query answer
Q

Fig. 3: The workflow for answering SIS$^{/\mathrm{MD}}$ queries.

As a result of the semantic separation of the two levels, the computational complexity of answering $\mathrm{SIS}^{/\mathrm{MD}}$ queries carries over directly from the complexity of answering queries in the two component languages. More specifically, if $(q, \mathcal{L})$ is the problem of answering queries of type q in the language $\mathcal{L}$ then the complexity bounds for answering $\mathrm{SIS}^{/\mathrm{MD}}$ queries satisfy the constraint:

$$\begin{gathered}\mathit{complexity}(m(y) : q(\overline{x}), \mathrm{SIS}^{/\mathrm{MD}}) = \\ \max(\mathit{complexity}(m(y), \mathcal{L}_{\mathcal{M}}), \mathit{complexity}(q(\overline{x}), \mathcal{L}_{\mathcal{O}}))\end{gathered}$$

For instance, if the meta-ontology and the object ontologies of a $\mathrm{SIS}^{/\mathrm{MD}}$ are expressed in OWL 2 DL, then answering queries expressible as OWL 2 DL class descriptions is N2ExpTime-complete, as this is the complexity of instance retrieval for concepts in the underlying DL $\mathcal{SROIQ}$ [8]. Although in principle the complexity of reasoning remains in the same class, in practice the actual effort of answering the object query can be dramatically reduced as the meta-query can significantly restrict the amount of data to be queried over.

Finally, we illustrate the querying mechanism with a small example involving the $\mathrm{SIS}^{/\mathrm{MD}}$ introduced in the previous section.

Example (cont.) Let $m_i(y) : q(x)$ be a $\mathrm{SIS}^{/\mathrm{MD}}$ query over $\mathcal{S}$, where we specify the object-query $q(x)$ as:

$$q(x) \leftarrow \exists y, z.(\boldsymbol{c}{:}\mathit{diagnosedWith}(x, y) \wedge \boldsymbol{d}{:}\mathit{disorderOf}(y, z) \wedge \boldsymbol{d}{:}\mathit{Heart}(z)),$$

and further, we vary the meta-query $m_i(y)$ across the following alternatives:

- $m_1(y) \leftarrow \boldsymbol{PatientKB}(y)$,
- $m_2(y) \leftarrow \boldsymbol{PatientKB}(y) \vee \boldsymbol{BimodicalKnowledge}(y)$,
- $m_3(y) \leftarrow \boldsymbol{PatientKB}(y) \vee \boldsymbol{BimodicalKnowledge}(y) \vee \boldsymbol{Mappings}(y)$.

Observe that different meta-queries return different sets of context names. Consequently, different subsets of object ontologies are selected as the basis for answering the object query. This in turn leads to obtaining different answers, as presented below:

meta-query	selected ontologies	object-query answers
$m_1(y)$	$\{\mathcal{O}_{\boldsymbol{c}}\}$	$Q = \emptyset$
$m_2(y)$	$\{\mathcal{O}_{\boldsymbol{c}}, \mathcal{O}_{\boldsymbol{d}}, \mathcal{O}_{\boldsymbol{e}}\}$	$Q = \{\boldsymbol{c}{:}\mathit{johnSmith}\}$
$m_3(y)$	$\{\mathcal{O}_{\boldsymbol{c}}, \mathcal{O}_{\boldsymbol{d}}, \mathcal{O}_{\boldsymbol{e}}, \mathcal{O}_{\boldsymbol{f}}\}$	$Q = \{\boldsymbol{c}{:}\mathit{johnSmith}, \boldsymbol{g}{:}\mathit{jSmith}\}$

3 Implementation

We implemented the $\mathrm{SIS}^{/\mathrm{MD}}$ framework using LarKC [2]; a platform for the creation and execution of Semantic Web reasoning workflows. Each LarKC workflow consists of a number of plugins. Each plugin performs some reasoning service over a given set of RDF statements. The platform ships with a number of pre-built plugins for various kinds of reasoning. Importantly, it has facilities for the development of reasoning plug-ins. Plug-ins can take advantage of a number of services available in the platform including execution on cluster machines and RDF data management. The SIS$^{/MD}$ framework was instantiated as LarKC workflow. Interestingly, the implemented workflow follows directly from the reasoning framework. The workflow consists of the following steps:

1. The metadata ontology is loaded into LarKC.
2. A SPARQL query representing the meta-ontology query is performed to select a series of files (i.e. knowledge sources) to be loaded. These files correspond to the object ontology. Note that the underlying triple store (OWLIM)[1] is configured to perform $pD*$ reasoning (also called OWL-HORST) [9] at this stage.
3. The selected files are loaded into LarKC. Note that under the expressive limitations of the OWL-HORST fragment the reduction step outlined in Table 2 can be omitted with little harm to the integration process.
4. A SPARQL query representing the object ontology query is performed and results are returned. Again, results are returned under OWL-HORST reasoning.

The workflow required only lightweight implementation of two LarKC plugins and the definition of the overall workflow. Importantly, only reasoning services that were already available in LarKC were required for the implementation of the framework. The workflow and associated plugins are accessible on-line at `http://www.few.vu.nl/~pgroth/sismd/`.

4 Case Study: Wordnet Alignment

We now describe the application of the framework to the reasoning over alignments between two versions of Wordnet: a large lexical database categorizing English words in linguistic categories.

4.1 Use Case

This use case stems from a cultural heritage portal serving documents that have been semantically annotated using Wordnet [10]. Because the portal integrates documents from different collections, part of the collection has been annotated using W3C's RDF representation of Wordnet 2.0, while another part uses 3.0. Obviously, one would like to be able to ignore the version differences when these are not relevant. For example, for a given query, all relevant documents annotated using either version need to be found.

To achieve this, an alignment needs to be created that describes, for as many concepts as possible, which concept in one version corresponds to the same or at least a very similar concept in the other version. Creating such an alignment is, however, not an exact science. In an idealized world, two concepts are either

[1] `http://www.ontotext.com/owlim`

equivalent or they are not. In practice, similarity levels vary on a continuous scale, and what level is "sufficiently similar" may depend on the application context. Additionally, similarity levels can vary on multiple dimensions. For example, a concept A can be very similar to B along one dimension, but more similar to C along another. A good weighing scheme that takes this into account is typically also application or context dependent. Finally, for large vocabularies such as Wordnet (both versions have over 100k concepts), the number of potential mappings (e.g. the Cartesian product of both sets) is very large, and automatic tools are needed to either fully automate the alignment process, or at least to help human experts in creating alignments interactively. As some correspondences are much harder to find than other, the resulting set of all correspondences produced by alignment tools tend to vary in nature and quality. An application might prefer to use only parts of the results.

Typically, creating a good alignment is a task that is too complex to be done at query time. On the other hand, alignments are too context-dependent to create a single alignment *a priori*. One solution is to create multiple sets of correspondences and annotate each set describing its properties. Applications can then query on a meta-level which sets are available and what their properties are. Based on this information, context-specific reasoning can be applied to decide which correspondences the system should use when answering future object-level queries. For example, a retrieval application might opt for high recall performance and include all mappings. An application that uses the mappings to upgrade a corpus that has been manually annotated would prefer high precisions alignments.

4.2 Alignment selection with SIS$^{/\mathrm{MD}}$

The described use-case is a typical example for an application with a large variety of related, but different sets of object-knowledge, together with a rich metadata ontology. We have applied the framework to Wordnet alignments produced using the Amalgame system [11]. To demonstrate its usage, we discuss a small example. A user is interested in how verbs can be aligned between Wordnet 3.0 and Wordnet 2.0. However, in one case the application is interested in mappings produced with the best numeric score as returned by the mapping algorithms. In the second case, the user is interested in mappings that were returned by multiple different mapping algorithms. Here, we show how modifying the meta-level query over the provenance changes the results of the same object-level query. For readability, we constrain the query to look at the word "catch".

Formally, we formulate two SIS$^{/\mathrm{MD}}$ queries: $m1(y) : q(e1, e2)$ and $m2(y) : q(e1, e2)$, with the same object query, looking for the pairs of alignment entities where one of them is of type VerbSynset and has a label "catch":

$$\begin{aligned} q(e1, e2) \leftarrow \exists x.(&align{:}entity1(x, e1) \wedge align{:}entity2(x, e2) \wedge \\ &wn20schema{:}VerbSynset(e1) \wedge label(e1, "catch")) \end{aligned}$$

```
select ?e1 ?e2 where {
    ?map align:entity1 ?e1.
    ?map align:entity2 ?e2.
    ?e1 rdf:type wn20sch:VerbSynset.
    ?e1 rdfs:label "catch"@en-us. }
```

Fig. 4: An example object-level query $q(e1, e2)$ formulated in SPARQL

The two variants of the meta-query used in the $\mathrm{SIS}^{/\mathrm{MD}}$ queries:

$$m1(y) \leftarrow \exists x.((\boldsymbol{wasGeneratedBy}(y, x) \wedge \boldsymbol{BestNumeric}(x)) \vee \boldsymbol{WordNetItem}(y))$$
$$m2(y) \leftarrow \exists x.((\boldsymbol{wasGeneratedBy}(y, x) \wedge \boldsymbol{MostMethods}(x)) \vee \boldsymbol{WordNetItem}(y))$$

define the relevant object data sources including those of type WordNetItem (effectively, the Wordnet ontologies) and all sources (the sets of mappings) generated with the BestNumeric and MostMethods approaches, respectively.

For this application, the object-level query $q(e1, e2)$ is formulated in SPARQL as shown in Figure 4. For all SPARQL queries, the following prefixes are defined: rdfs:http://www.w3.org/2000/01/rdf-schema#, rdf:http://www.w3.org/1999/02/22-rdf-syntax-ns#, ag:http://purl.org/vocabularies/amalgame#, opmv:http://purl.org/net/opmv/ns#, wn20sch:http://www.w3.org/2006/03/wn/wn20/schema/, align:http://knowledgeweb.semanticweb.org/heterogeneity/alignment#

```
select ?file where {
 {?file opmv:wasGeneratedBy ?alg.
  ?alg rdf:type ag:BestNumeric. }
 UNION
 {?file rdf:type ag:WordNetItem.}}
```

Fig. 5: An example meta-level query $m1(x)$ formulated in SPARQL.

```
select ?file where {
  {?file opmv:wasGeneratedBy ?s.
   ?s rdf:type ag:MostMethods.}
  UNION
  {?file rdf:type ag:WordNetItem.}}
```

Fig. 6: An example meta-level query $m2(x)$ formulated in SPARQL.

The meta-level query $m1(x)$ for best-numeric mappings algorithms is shown in Figure 5. Applying, the object-level query over the results of $m1(x)$ (446 377 triples), produces two result bindings mapping the same synset:

```
Binding 1 and 2:
?e1: http://purl.org/vocabularies/princeton/wn30/synset-catch-verb-18
?e2: http://www.w3.org/2006/03/wn/wn20/instances/synset-catch-verb-18
```

The meta-level query $m2(x)$ for mappings from different mapping algorithms is shown in Figure 6. With this query, 353 303 triples are used and no results are returned. Note, that here the mappings are only 167 triples of the total number of triples as compared to 93 241 triples for the prior set of mappings. While the above queries are simple, they do require reasoning. They show how by changing the view over provenance (or meta information) systems can achieve different results. Most importantly, the case study emphasizes the simplicity of the framework and the ease with which it can be implemented and applied.

5 Related Work

The SIS$^{/\mathrm{MD}}$ framework is a strongly restricted fragment of Context Description Logics introduced in [12,13], which are based on combinations of pairs of Description Logics. The restriction concerns the number of contexts (boxes) allowed (here finite, while possibly unbounded in the general case) and the expressiveness of the interoperability mechanism. This fragment also closely coincides with the architecture of Contextualized Knowledge Repositories (CKRs), proposed in [6]. The notable difference is that the meta-language in CKRs is pruned down to a fixed set of contextual properties, e.g: time, location, topic, along with their pre-defined values and the coverage relation for organizing contexts in a generality-specificity hierarchy. In contrast, the meta-language in SIS$^{/\mathrm{MD}}$ is an arbitrary, unrestricted DL language, whose vocabulary and expressiveness are left entirely as an application-driven choice. In the pure RDF paradigm, another framework similar to ours and CKRs, called RDF$^+$ is discussed in [14] and based on the use of Named Graphs [15] for representing both levels of knowledge. Given the expressive limitations of RDF, the scope of meta-language in RDF$^+$ is again restricted to a set of relational properties. Moreover, unlike in our case, the notions of selection and integration of the object-level knowledge are not considered. A framework that supports meta-level selection of object-level knowledge was proposed in [16]. It provides a mechanism for selecting a subset of a single ontology based on axioms annotations. The framework, however, does not support the context-sensitive integration, in the sense discussed here, as it is assumed that the entire object-level knowledge is given in one ontology.

The semantic interoperability mechanism involved in our framework is based directly on the Simple Interoperability Systems, defined in [13], and as argued there, it remains a notational variant of the Package-based DLs (P-DLs) [7]. Consequently, it bares certain similarities to other logic-based ontology integration formalisms [17], such as e.g. Distributed DLs [18] or $\mathcal{E}$-Connections [19]. Differently than in our case, the interoperability in those two formalisms is achieved by use of external bridge rules or internal link relations connecting the vocabularies of different ontologies. Such constructs are then interpreted in terms of mappings between the models of the connected ontologies. Such an approach grants a weaker style of integration (less inferences possible) but a more robust one with respect to possible inconsistencies arising due to heterogeneity of integrated knowledge. For a formal survey, we refer the reader to [17].

6 Conclusion

We presented a framework that allows for the adaptive selection and integration of object-level knowledge based on meta-level knowledge. To the best of our knowledge, this is the first formal framework that deals with the interrelationship between meta-level knowledge and object-level knowledge purely in terms of standard Semantic Web knowledge representations (e.g. Description Logics, OWL and RDFS). Importantly, we demonstrated that the framework can be realized using an existing Semantic Web development framework (LarKC) and applied to an existing use case; the alignment of vocabularies in a cultural heritage setting. Going forward, we aim to study the application of the framework in more dynamic or streaming settings. Additionally, we aim to apply the approach to large sets of biomedical concept mappings provided by a range of providers.

References

1. Tummarello, G., Cyganiak, R., Catasta, M., Danielczyk, S., Delbru, R., Decker, S.: Sig.ma: Live views on the web of data. J. Web Sem. **8**(4) (2010) 355–364
2. Fensel, D., van Harmelen, F., Andersson, B., Brennan, P., Cunningham, H., Della Valle, E., Fischer, F., Huang, Z., Kiryakov, A., il Lee, T.K., School, L., Tresp, V., Wesner, S., Witbrock, M., Zhong, N.: Towards larkc: a platform for web-scale reasoning. In: Proceedings of the IEEE International Conference on Semantic Computing (ICSC 2008), August 4-7, 2008, Santa Clara, CA, USA. (8 2008)
3. Horrocks, I., Patel-schneider, P.F., Harmelen, F.V.: From SHIQ and RDF to OWL: The making of a Web Ontology Language. Journal of Web Semantics **1** (2003) 2003
4. Motik, B.: Owl 2 web ontology language profiles. Technical report, W3C Recommendation: `http://www.w3.org/TR/owl2-profiles/` (2009)
5. Baader, F., Calvanese, D., Mcguinness, D.L., Nardi, D., Patel-Schneider, P.F.: The description logic handbook: theory, implementation, and applications. Cambridge University Press (2003)
6. Homola, M., Serafini, L., Tamilin, A.: Modeling contextualized knowledge. In: Procs. of the 6th Workshop on Semantic Web Applications and Perspectives (SWAP2010). (2010)
7. Bao, J., Voutsadakis, G., Slutzki, G., Honavar, V.: Package-based description logics. In Stuckenschmidt, H., Parent, C., Spaccapietra, S., eds.: Modular Ontologies. (2009) 349–371
8. Kazakov, Y.: Riq and sroiq are harder than shoiq. In: In Proc. KR08. (2008)
9. ter Horst, H.J.: Completeness, decidability and complexity of entailment for rdf schema and a semantic extension involving the owl vocabulary. Web Semant. **3** (October 2005) 79–115
10. Schreiber, G., Amin, A., Aroyo, L., van Assem, M., de Boer, V., Hardman, L., Hildebrand, M., Omelayenko, B., van Osenbruggen, J., Tordai, A., Wielemaker, J., Wielinga, B.: Semantic annotation and search of cultural-heritage collections: The Multimedian E-Culture demonstrator. Journal of Web Semantics **6**(4) (2008) 243–249
11. Jacco van Ossenbruggen, M.H., de Boer, V.: Interactive vocabulary alignment. In: Proceedings of the International Conference on Theory and Practice of Digital Libraries (TPDL 2011), Berlin (September 2011)
12. Klarman, S., Gutiérrez-Basulto, V.: $\mathcal{ALC}_{\mathcal{ALC}}$: A context description logic. In: Proc. of the European Conference on Logics in Artificial Intelligence. (2010)
13. Klarman, S., Gutiérrez-Basulto, V.: Two-dimensional description logics for context-based semantic interoperability. In: Proc. of AAAI-11. (2011)
14. Dividino, R., Sizov, S., Staab, S., Schueler, B.: Querying for provenance, trust, uncertainty and other meta knowledge in rdf. Journal of Web Semantantics **7** (September 2009) 204–219
15. Carroll, J.J., Bizer, C., Hayes, P., Stickler, P.: Named graphs. Journal of Web Semantics **3** (2005) 247–267

16. Tran, T., Haase, P., Motik, B., Grau, B.C., Horrocks, I.: Metalevel information in ontology-based applications. In: Proc. of AAAI-08. (2008)
17. Cuenca Grau, B., Kutz, O.: Modular ontology languages revisited. In: Proc. of the Workshop on Semantic Web for Collaborative Knowledge Acquisition. (2007)
18. Borgida, A., Serafini, L.: Distributed description logics: Assimilating information from peer sources. Journal of Data Semantics **1** (2003) 2003
19. Cuenca Grau, B., Parsia, B., Sirin, E.: Modular ontologies. Springer-Verlag, Berlin, Heidelberg (2009) 293–320