Typed Higher-Order Variant of SROIQ – Why Not?

Martin Homola[1], Ján Kľuka[1], Vojtěch Svátek[2], and Miroslav Vacura[2]

[1] Comenius University in Bratislava
Mlynská dolina, 842 48 Bratislava, Slovakia
[2] University of Economics, Prague
W. Churchill Sq. 4, 130 67 Prague 3, Czech Republic

Abstract. We provide $\mathcal{TH}(\mathcal{SROIQ})$ a typed higher-order extension of $\mathcal{SROIQ}$, that is itself a syntactic variant of $\mathcal{SROIQ}$, i.e., it is polynomially reducible into it. We also discuss modelling scenarios in which such a language is useful.

1 Introduction

Metamodelling, i.e., logical representation of relations (other than subsumption) involving concepts and roles, including their membership in meta-concepts, has been recently subject to an increasing number of research proposals [16,18,15,3,4]. Different applications are cited, roughly divisible into two categories: (i) modelling of certain complex domains, e.g., classifying taxonomic biological concepts such as GoldenEagle and Accipitriformes into higher-order concepts such as Species, Order, etc. [4], or, similarly, classifying GoldenEagle into EndangeredSpecies [15]; or, further, modelling the topic relation, for which values as abstract as whole concepts are sometimes considerable – a book can be about eagles, hence assigning the concept GoldenEagle as a topic may sometimes make sense [16]; and (ii) expressing meta-properties of the modelled entities themselves, such as deprecation of some concepts by assigning them into the Deprecated meta-concept, or identification of authors of model entities (concepts, roles) using the meta-role definedBy [3], or sorting concepts into diagnostic meta-concepts according to some ontology analysis methodology (e.g., OntoClean [6]) [4].

Metamodelling is not enabled by DL or OWL, however in many ontologies we find hints that it could be useful. In Fig. 1 (a) we can see a usage of the role mo:primary_instrument from Music Ontology (MO) [21]. Nigel Kennedy's primary instrument is violin, represented by the individual mo-mit:Violin. This individual in reality represents the notion of violin as a musical instrument type, hence it's more a "concept" represented by an individual. Indeed we would assert ex:Stradivarius_1707 as its instance.

Using classes as property values is not allowed in OWL, hence we need individuals such as mo-mit:Violin. More examples can be seen in Fig. 1 (b): in MO particular CDs (i.e., "true" individuals) are assigned as members to their release by frbr:exemplar_of. A release not just groups individuals released by a publisher at a given time, but also abstracts their common features (i.e, it is more like a concept in this respect). The release "concepts" are categorized into several types (singles, albums, etc.), which in MO are grouped under mo:ReleaseType (i.e, these are already "meta-concepts" in certain sense).

Note that we do not suggest that the actual modelling of MO is wrong; in fact it functions well as it is. We are saying that in MO, and in many other Linked Data (LD)
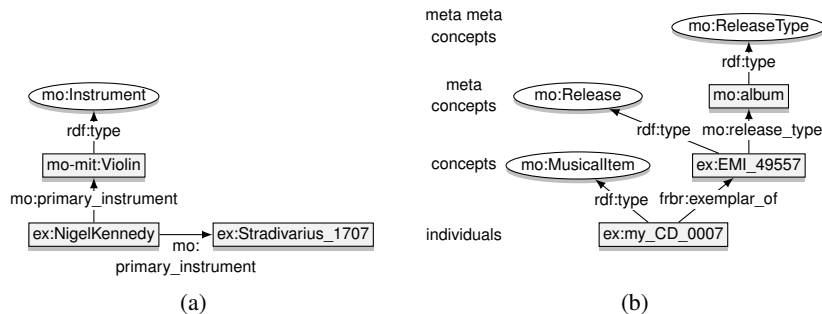
Fig. 1. Music Ontology excerpts: (a) primary instrument issue; (b) recognizable higher orders

vocabularies, opportunities for more widespread applications of metamodelling can be found, if it was enabled in OWL. It could provide for more obvious models, clearer separation of levels of abstraction, and further logical consequences, as we exemplify it later on (see Sect. 4). From the logical point of view, metamodelling amounts to modelling with higher orders. To enable it, DL and OWL have to be extended with such constructs, as already proposed by a number of researchers [15,3,4]. However, this extension needs to be done in a way that is useful and intuitive to the users. We suggest the following properties to be followed:

– Clear separation between individuals, roles, concepts, and meta-entities of various kinds, in order to avoid confusing models where one entity has multiple rôles.
– Intuitive semantic properties, especially a reasonable relation between intension and extension of concepts to avoid paradoxical and unexpected consequences.
– Avoiding unreasonable computational complexity jumps.

According to our opinion (see further discussion in Sect. 5) none of the existing approaches yet got thus far. We therefore propose a higher-order extension of $\mathcal{SROIQ}$ [12] – the DL behind OWL 2 [17] – in which concepts may be used as individuals (as property values, assigned into meta-concepts, etc.) but which still keeps a strict separation between concepts and roles. The logic, dubbed $\mathcal{TH}(\mathcal{SROIQ})$, enables also meta-meta-concepts, and so on, ordered into a hierarchy of higher types. It employs a HiLog-style semantics [1], already used by RDF [7] and likewise other higher-order DL proposals [15,3]. This semantics fulfils some reasonable properties, as we discuss. We further show by reduction that $\mathcal{TH}(\mathcal{SROIQ})$ is decidable and in fact its computational complexity is the same as of regular $\mathcal{SROIQ}$. Finally in the second part of the paper we illustrate some modelling scenarios in which our language can be useful. Proofs are included in the extended version of the paper [10].

2 Preliminaries

$\mathcal{SROIQ}$ [12] uses a DL vocabulary $N = N_C \uplus N_R \uplus N_I$ with countable sets of *atomic concepts* (N_C), *atomic roles* (N_R), and *individuals* (N_I). *Complex concepts* (*complex*

Table 1. Syntax and Semantics of $\mathcal{SROIQ}$

Cons'tor	Syntax	Semantics		Syn.	Semantics
complement	$\neg C$	$\Delta^I \setminus C^I$	nominal	$\{a\}$	$\{a^I\}$
intersection	$C \sqcap D$	$C^I \cap D^I$	inv. role	R^-	$\{\langle y,x \rangle \mid \langle x,y \rangle \in R^I\}$
exist. restr.	$\exists R.C$	$\{x \mid \exists y\, \langle x,y \rangle \in R^I \wedge y \in C^I\}$	univ. role	U	$\Delta^I \times \Delta^I$
card. restr.	$\geqslant nR.C$	$\{x \mid \sharp\{y \mid \langle x,y \rangle \in R^I \wedge y \in C^I\} \geq n\}$	chain (w)	$S{\cdot}Q$	$S^I \circ Q^I$
self restr.	$\exists R.\mathrm{Self}$	$\{x \mid \langle x,x \rangle \in R^I\}$			

Axiom	Syntax	Semantics		Syntax	Semantics
conc. incl. (GCI)	$C \sqsubseteq D$	$C^I \subseteq D^I$	concept assert.	$a\colon C$	$a^I \in C^I$
role incl. (RIA)	$w \sqsubseteq R$	$w^I \subseteq R^I$	role assertion	$a,b\colon R$	$\langle a^I,b^I \rangle \in R^I$
reflexivity assert.	$\mathrm{Ref}(R)$	R^I is reflexive	neg. role assert.	$a,b\colon \neg R$	$\langle a^I,b^I \rangle \notin R^I$
role disjointness	$\mathrm{Dis}(P,R)$	$P^I \cap R^I = \emptyset$			

roles) are defined as the smallest sets containing all concepts and roles that can be inductively constructed using the concept (role) constructors[3] in Table 1, where C, D are concepts, R is an atomic or inverse role, S, Q are any roles (including role chains), a, b are individuals, and n is a positive integer. A $\mathcal{SROIQ}$ *knowledge base* (KB) is a finite set $\mathcal{K}$ of *axioms* of the types shown in Table 1.

A DL *interpretation* is a pair $\mathcal{I} = \langle \Delta^I, \cdot^I \rangle$ where $\Delta^I \neq \emptyset$ and $\cdot^I$ is a function such that $a^I \in \Delta^I$ for all $a \in N_I$, $A^I \subseteq \Delta^I$ for all $A \in N_C$, and $R^I \subseteq \Delta^I \times \Delta^I$ for all $R \in N_R$. Interpretation of complex concepts and roles is inductively defined as given in Table 1. An axiom φ is *satisfied* by $\mathcal{I}$ (denoted $\mathcal{I} \models \varphi$) if $\mathcal{I}$ satisfies the respective semantic constraint listed in Table 1, and $\mathcal{I}$ is a *model* of $\mathcal{K}$ (denoted $\mathcal{I} \models \mathcal{K}$) if it satisfies all axioms of $\mathcal{K}$. A concept C is *satisfiable w.r.t.* $\mathcal{K}$ if there is a model of $\mathcal{K}$ such that $C^I \neq \emptyset$. A formula φ is *entailed by* $\mathcal{K}$ (denoted $\mathcal{K} \models \varphi$) if $\mathcal{I} \models \varphi$ for all models $\mathcal{I}$ of $\mathcal{K}$. The two reasoning tasks of *concept satisfiability* and *concept subsumption* entailment are inter-reducible [12]. Reasoning in $\mathcal{SROIQ}$ is decidable if further syntactic restrictions are applied [12], which are based on the following notions.

A role R is *simple*, if (a) it is atomic and it does not occur on the right-hand side of any RIA; (b) it is an inverse of a simple role S; (c) if occurs on the right-hand side of a RIA but each such RIA is of the from $S \sqsubseteq R$ where S is a simple role.

A regular order on roles $\prec$ is a strict partial order (transitive and irreflexive binary relation) on roles (including inverse roles) such that $S \prec R$ iff $S^- \prec R$ for any roles S, R. Given a regular order on roles $\prec$, a RIA is $\prec$-*regular* if it has one of the following forms: (a) $R{\cdot}R \sqsubseteq R$; (b) $R^- \sqsubseteq R$; (c) $S_1 \cdot \cdots \cdot S_n \sqsubseteq R$ and $S_i \prec R$ for $1 \leq i \leq n$; (d) $R{\cdot}S_1 \cdot \cdots \cdot S_n \sqsubseteq R$ and $S_i \prec R$ for $1 \leq i \leq n$; (e) $S_1 \cdot \cdots \cdot S_n{\cdot}R \sqsubseteq R$ and $S_i \prec R$ for $1 \leq i \leq n$.

Finally, the restrictions placed on $\mathcal{SROIQ}$: (a) only simple roles may appear in cardinality restrictions; self restriction, and role disjointness axioms; (b) there is a regular order of roles $\prec$ such that all RIAs in the KB are $\prec$-regular; (c) the universal role may not appear in any RIA, role reflexivity, nor disjointness axiom.

Under these restrictions satisfiability (and thus subsumption) is decidable [12] and it is 2-NExpTime complete [13].

---

[3] There are additional $\mathcal{SROIQ}$ constructors and axioms all of which (including equality) can be reduced to the core constructs listed in Table 1.

Table 2. Syntax and HiLog-Style Semantics of $\mathcal{TH}(\mathcal{SROIQ})$

tu-role expr. (ρ^{tu})	Extension ($\rho^{tu\mathcal{E}}$)	ρ^{tu}	$\rho^{tu\mathcal{E}}$
R^{tu}	$R^{tu\,\mathcal{I}\mathcal{E}}$	R^{ut-}	$\{\,\langle y,x\rangle \mid \langle x,y\rangle \in R^{ut\mathcal{E}}\,\}$
U^{tu}	$\Delta^{\mathcal{I}}_{t-1} \times \Delta^{\mathcal{I}}_{u-1}$	$S^{tv}{\cdot}Q^{vu}\ (w^{tu})$	$S^{tv\mathcal{E}} \circ Q^{vu\mathcal{E}}$

t-description (γ^t)	Extension ($\gamma^{t\mathcal{E}}$)	γ^t	$\gamma^{t\mathcal{E}}$
A^t	$A^{t\,\mathcal{I}\mathcal{E}}$	$\exists R^{tu}.C^u$	$\{\,x \mid \exists y.\langle x,y\rangle \in R^{tu\mathcal{E}} \wedge y \in C^{u\mathcal{E}}\,\}$
$\neg C^t$	$\Delta^{\mathcal{I}}_{t-1} \setminus C^{t\mathcal{E}}$	$\geqslant n S^{tu}.C^u$	$\{\,x \mid \sharp\{\,y \mid \langle x,y\rangle \in S^{tu\mathcal{E}} \wedge y \in C^{u\mathcal{E}}\,\} \geq n\,\}$
$C^t \sqcap D^t$	$C^{t\mathcal{E}} \cap D^{t\mathcal{E}}$	$\exists S^{tt}.\mathsf{Self}$	$\{\,x \mid \langle x,x\rangle \in S^{tt\mathcal{E}}\,\}$
$\{A^{t-1}\}$	$\{A^{t-1\,\mathcal{I}}\}$		

Axiom (φ)	Semantics ($\mathcal{I} \models \varphi$)	φ	$\mathcal{I} \models \varphi$
$C^t \sqsubseteq D^t$	$C^{t\mathcal{E}} \subseteq D^{t\mathcal{E}}$	$w^{tu} \sqsubseteq S^{tu}$	$w^{tu\mathcal{E}} \subseteq S^{tu\mathcal{E}}$
$A^{t-1} : C^t$	$A^{t-1\,\mathcal{I}} \in C^{t\mathcal{E}}$	$\mathsf{Ref}(R^{tt})$	$R^{tt\mathcal{E}}$ is reflexive
$A^{t-1}, B^{u-1} : R^{tu}$	$\langle A^{t-1\,\mathcal{I}}, B^{u-1\,\mathcal{I}}\rangle \in R^{tu\mathcal{E}}$	$\mathsf{Dis}(R^{tu}, S^{tu})$	$R^{tu\mathcal{E}} \cap S^{tu\mathcal{E}} = \emptyset$
$A^{t-1}, B^{u-1} : \neg R^{tu}$	$\langle A^{t-1\,\mathcal{I}}, B^{u-1\,\mathcal{I}}\rangle \notin R^{tu\mathcal{E}}$		

3 Typed Higher-Order $\mathcal{SROIQ}$

3.1 Syntax

We introduce $\mathcal{TH}(\mathcal{SROIQ})$, the typed higher-order variant of $\mathcal{SROIQ}$. For any sub-language $\mathcal{L}$ of $\mathcal{SROIQ}$, its typed higher-order variant $\mathcal{TH}(\mathcal{L})$ is derived analogously. The language is constructed over a vocabulary which explicitly distinguishes between entities of different types.

Definition 1 (Typed DL Vocabulary). *A typed DL vocabulary N is a disjoint union of a countable number of countable sets of the form $\biguplus_{t \geq 0} N^t_{\mathrm{C}} \uplus \biguplus_{t > 0, u > 0} N^{tu}_{\mathrm{R}}$ where for each $t \geq 0$, N^t_{C} is the set of* concept names of order t, *and for each $t, u > 0$, N^{tu}_{R} is the set of* role names between orders t and u. *The set N^0_{C} is also called the set of* individual names *and denoted by N_{I}. The* type *of a name $X \in N$ is either the number t if X is a concept name of order t, or the pair of numbers tu if X is a role name between orders t and u.*

For clarity of presentation, we introduce the following notation: Symbols of the form $A^t, B^t, \ldots$ belong to N^t_{C}, and specifically, symbols of the form $A^0, B^0, \ldots, a, b, \ldots$ belong to N_{I}. Symbols of the form $R^{tu}, S^{tu}, \ldots$ belong to N^{tu}_{R}. Similarly for longer symbols, e.g., $\mathsf{NigelKennedy} \in N^0_{\mathrm{C}}$, $\mathsf{InstrumentType}^2 \in N^2_{\mathrm{C}}$ and $\mathsf{primaryInstrument}^{12} \in N^{12}_{\mathrm{R}}$.

Complex concepts and roles are also typed, and can only be combined in ways allowed in the following definition.

Definition 2 (tu-role expression, t-description). *The sets of tu-role expressions for all $t, u > 0$ are simultaneously inductively defined as the smallest sets containing the expressions listed in the upper part of Table 2, where $R^{tu} \in N^{tu}_{\mathrm{R}}$, $v > 0$, S^{tv} is a tu-role expression and Q^{vu} a vu-role expression respectively. We also define $(R^{tu-})^- = R^{tu}$ for any $R^{tu} \in N^{tu}_{\mathrm{R}}$.*

The sets of t-descriptions for all $t > 0$ are simultaneously inductively defined as the smallest set containing the expressions listed in the middle part of Table 2, where $t, u > 0$, $A^t \in N^t_{\mathrm{C}}$, $A^{t-1} \in N^{t-1}_{\mathrm{C}}$, C^t and D^t are t-descriptions, C^u is an u-description, and R^{tu} (R^{tt}) is an atomic or inverse tu-role (tt-role).

The typing restrictions are quite straightforward: The inverse of a role swaps its source and target orders, a role chain connects roles with matching target and source orders. Only concepts of the same order can be intersected. Restrictions on *tu*-roles construct descriptions of order t if target descriptions are of order u (or $t = u$ for the self restriction). The nominal constructs a singleton t-concept on top of a $(t-1)$-concept. Additional concept constructors ($\sqcup$, $\forall$, $\leqslant$) can be derived from those listed in Table 2 just as in the first-order $\mathcal{SROIQ}$. Note that there is a separate universal role U^{tu} for each pair of orders $t, u > 0$, and for each order $t > 0$, a separate top (bottom) concept $\top^t$ ($\bot^t$) can be introduced as an abbreviation of $A^t \sqcup \neg A^t$ ($A^t \sqcap \neg A^t$) using some $A^t \in N_C^t$.

Finally, $\mathcal{TH}(\mathcal{SROIQ})$ KBs are defined as follows.

Definition 3 (Knowledge base). *A $\mathcal{TH}(\mathcal{SROIQ})$ knowledge base $\mathcal{K}$ is a finite set of axioms of the forms listed in the bottom part of Table 2, where $t, u > 0$, A^{t-1} and B^{u-1} are atomic concepts of N_C^{t-1} and N_C^{u-1}, respectively, C^t and D^t are t-descriptions, R^{tu} and S^{tu} are tu-role expressions, R^{tt} is a tt-role expressions, and w^{tu} is a tu-role chain.*

We can again see that when an axiom relates two concepts or role expressions, their types must match exactly. On the other hand, the last three axioms, responsible for concept and role membership, assign concepts of a lower order to concepts and roles exactly one order higher. This is a fundamental principle of the type hierarchy (which also exactly reflected by the nominal constructor as discussed above).

It turns out that in order to assure decidability of $\mathcal{TH}(\mathcal{SROIQ})$ – as we will prove later on – the same syntactic restriction can be applied as in regular $\mathcal{SROIQ}$: (a) only simple roles may appear in cardinality restrictions, self restriction, and role disjointness axioms; (b) all RIAs in the KB are $\prec$-regular for some regular order of roles $\prec$; (c) the universal role may not appear in RIAs, role reflexivity, nor disjointness axioms. Simple roles and $\prec$-regular RIAs are defined exactly as in the previous section.

3.2 Semantics

Typed higher-order description logics can be given several semantics, depending on the intended use. In our previous paper [9], we have discussed two options: one based on Henkin's general models [8] of the simple theory of types, and another based on the HiLog semantics [1] of higher-order logic programming. Both options have been considered in the DL context before (see Sect. 5). Henkin's semantics, although more straightforward, is not suitable for applications discussed below in Sect. 4. This is due to its extensionality property requiring two concepts to be equal if they have equal sets of instances. We will thus only consider the HiLog-style semantics in this paper.

HiLog-style interpretations rely on a single domain $\Delta^{\mathcal{I}}$. They first assign each name an element of $\Delta^{\mathcal{I}}$ serving as the name's *intension*, for which we will use the function $\cdot^{\mathcal{I}}$. The intension is then possibly assigned a *concept extension* (a subset of the domain $\Delta^{\mathcal{I}}$) and a *role extension* (a subset of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$), for which we will use the function $\cdot^{\mathcal{E}}$. We obtain a typed version of this semantics by partitioning the domain $\Delta^{\mathcal{I}}$ into disjoint slices, each dedicated to intensions of members of one type (either a concept type t, or a role type tu). Domain elements are then assigned extensions as appropriate, i.e., concept intensions are only assigned concept extensions, role intensions only role extensions.

Definition 4 (HiLog-Style Interpretation). *Let N be a typed DL vocabulary. A HiLog-style interpretation of N is a triple $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}}, \cdot^{\mathcal{E}})$ such that*

1. *$\Delta^{\mathcal{I}} = \biguplus_{t \geq 0} \Delta_t^{\mathcal{I}} \uplus \biguplus_{t,u > 0} \Delta_{tu}^{\mathcal{I}}$ is a disjoint union of countably many sets and $\Delta_0^{\mathcal{I}} \neq \emptyset$,*
2. *$A^{t\,\mathcal{I}} \in \Delta_t^{\mathcal{I}}$ for each $A^t \in N_{\mathsf{C}}^t$ and $t \geq 0$,*
3. *$R^{tu\,\mathcal{I}} \in \Delta_{tu}^{\mathcal{I}}$ for each $R^{tu} \in N_{\mathsf{R}}^{tu}$ and $t, u > 0$,*
4. *$c^{\mathcal{E}} \subseteq \Delta_{t-1}^{\mathcal{I}}$ for each $c \in \Delta_t^{\mathcal{I}}$ and $t > 0$,*
5. *$r^{\mathcal{E}} \subseteq \Delta_{t-1}^{\mathcal{I}} \times \Delta_{u-1}^{\mathcal{I}}$ for each $r \in \Delta_{tu}^{\mathcal{I}}$ and $t, u > 0$,*

and the interpretation of role expressions and complex descriptions is inductively defined according to, respectively, the upper and middle part of Table 2, where (and thenceforth), by abuse of notation, we redefine $X^{\mathcal{E}} := (X^{\mathcal{I}})^{\mathcal{E}}$ for atomic concepts and atomic roles, and $X^{\mathcal{E}} := X^{\mathcal{E}}$ for non-atomic ones.

Definition 5 (HiLog-Style Satisfaction). *Given a formula (axiom) φ in $\mathcal{TH}(\mathcal{L})$, a HiLog-style interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}}, \cdot^{\mathcal{E}})$ satisfies φ (denoted $\mathcal{I} \models \varphi$) under the conditions specified in the lower part of Table 2.*

Definition 6 (HiLog-Style Model). *A HiLog-style interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}}, \cdot^{\mathcal{E}})$ is a model of a $\mathcal{TH}(\mathcal{L})$ KB $\mathcal{K}$ if $\mathcal{I} \models \varphi$ for all $\varphi \in \mathcal{K}$.*

As remarked by Chen et al. [1], this way an essentially first-order semantics (free of higher-order power sets) for a higher-order language is obtained. On the other hand, we have to be careful to properly distinguish between the two interpretations $\cdot^{\mathcal{I}}$ and $\cdot^{\mathcal{E}}$.

As argued in our previous work [9], two important properties of a semantics for metamodelling are: *Intensional regularity* requires that if intensions of two names are equal ($\mathcal{K} \models A^t = B^t$), then so are their extensions ($\mathcal{K} \models A^t \equiv B^t$). *Extensionality*, found in set theory, is the converse of intensional regularity (if $\mathcal{K} \models A^t \equiv B^t$, then $\mathcal{K} \models A^t = B^t$). We have come to conclusion that for metamodelling applications where higher-order names are intended to represent concepts, intensional regularity is a desirable property (see also Example 2 in Sect. 4.1), but extensionality may cause counterintuitive results. The HiLog-style semantics is intensionally regular, but not extensional, which makes it especially suitable for such applications. Henkin semantics has both properties, which makes it too strong for metamodelling. (For more details see [9].)

3.3 Decidability and Reasoning

We will now demonstrate decidability of decision problems in $\mathcal{TH}(\mathcal{SROIQ})$ under the HiLog-style semantics, and examine their complexity. In our previous work [9], we have only obtained decidability for $\mathcal{TH}(\mathcal{ALCHOIQ})$. The new results for $\mathcal{TH}(\mathcal{SROIQ})$ are obtained by a reduction directly to first-order $\mathcal{SROIQ}$, which builds on ideas by Glimm et al. [4].

The first-order reduction matches the duality of intensions and extensions in the HiLog-style semantics by representing each concept A^t for $t > 0$ with two names. The concept name A^t represents the extension. The individual name c_{A^t} represents the intension and replaces A^t when used as an instance of higher-order concepts or a participant in higher-order roles. The relationship between the names A^t and c_{A^t} is expressed by a suitable axiom, which connects exactly the instances of A^t with c_{A^t} via a new instanceOf role, thus ensuring intensional regularity. Let us now define the reduction in detail.

Table 3. Components of the first-order reduction

Typing(N)

$\top \sqsubseteq \exists U.\top^1$		$\top^t \sqsubseteq \neg\top^u$ for each $t, u > 0$, $t \neq u$
$A^t \sqsubseteq \top^t$	$c_{A^t} : \top^{t+1}$ for each $A^t \in N_C^t$, $t > 0$	$A^0 : \top^1$ for each $A^0 \in N_C^0$, $t > 0$
$\top \sqsubseteq \forall R^{tu}.\top^u \quad \exists R^{tu}.\top \sqsubseteq \top^t$	for each $R^{tu} \in N_R^{tu}$, $t, u > 0$	

InstSync(N)

$A^t \equiv \exists \text{instanceOf}.\{c_{A^t}\}$	for each $A^t \in N_C^t$, $t > 0$

Definition 7 (First-Order Reduction). *Let $\mathcal{K}$ be a $\mathcal{TH}(\mathcal{SROIQ})$ KB using a vocabulary $N = \biguplus_{t \geq 0}\{N_C^t\} \uplus \biguplus_{t,u > 0}\{N_R^{tu}\}$. The vocabulary is reduced into a DL vocabulary $N^1 = (N_C, N_R, N_I)$ as follows:*

$$N_C = \biguplus_{t > 0} N_C^t \uplus \biguplus_{t > 0}\{\top^t\}, \quad N_R = \biguplus_{t,u > 0} N_R^{tu} \uplus \{\text{instanceOf}\}, \quad N_I = N_C^0 \uplus \biguplus_{t > 0}\{c_{A^t} \mid A^t \in N_C^t\}.$$

The names $\top^t$, instanceOf, and c_{A^t} are fresh, and type indices are parts of these names, not mere annotations. We call c_{A^t} the metamodelling individual *for A^t.*

The first-order reduction *of $\mathcal{K}$ is a $\mathcal{SROIQ}$ KB $\mathcal{K}^1$ over N^1, defined as follows: $\mathcal{K}^1 = \text{Bound}(\mathcal{K}) \cup \text{Typing}(N) \cup \text{InstSync}(N)$, where $\text{Typing}(N)$ and $\text{InstSync}(N)$ are defined in Table 3, and $\text{Bound}(\mathcal{K})$ is an extension to sets of the following transformation of $\mathcal{TH}(\mathcal{SROIQ})$ axioms:*

1. *each occurrence of a t-description C of the form $\neg D^t$, $\exists U^{tu}.D^u$, or $\geq nU^{tu}.D^u$ for any $n \geq 0$, $t, u > 0$ is replaced by $\top^t \sqcap C$;*
2. *each occurrence of U^{tu} is replaced by U;*
3. *each occurrence of $A^t \in N_C^t$ for $t > 0$ in a nominal, or in the left-hand side of a concept assertion, a (negative) role assertion is replaced by c_{A^t}.*

The reduction fully captures the HiLog-style semantics, or, more formally:

Theorem 1. *For any $\mathcal{TH}(\mathcal{SROIQ})$ KB $\mathcal{K}$ in a vocabulary N and any $\mathcal{TH}(\mathcal{SROIQ})$ axiom φ in N, we have $\mathcal{K} \models \varphi$ iff $\mathcal{K}^1 \models \text{Bound}(\varphi)$.*

The first-order reduction allows for decidability of $\mathcal{TH}(\mathcal{SROIQ})$ with the HiLog-style semantics in the same complexity class as first-order $\mathcal{SROIQ}$.

Corollary 1 *Concept satisfiability and subsumption in $\mathcal{TH}(\mathcal{SROIQ})$ with HiLog satisfiability are decidable in 2-NExpTime.*

Both proofs can be found in the extended version of the paper [10].

4 Modelling in $\mathcal{TH}(\mathcal{SROIQ})$

4.1 Basic Metamodelling

We will remodel the situation from Fig. 1 (a) in $\mathcal{TH}(\mathcal{SROIQ})$. Note that we will also slightly change the notation and omit namespaces for simplicity. We can now use individuals for musicians and for particular instruments, 1st-order concepts for instrument types, grouped under the 2nd-order concept InstrumentType[2]. See Example 1.

Example 1. The fragment in question now looks as follows:

$$\text{NigelKennedy}: \text{MusicArtist}^1 \qquad \text{Stradivarius\_1707}: \text{Violin}^1 \qquad \text{Violin}^1: \text{Instrument}^2$$
$$\text{NigelKennedy}, \text{Stradivarius\_1707}: \text{primaryInstrument}^{11}$$
$$\text{NigelKennedy}, \text{Violin}^1: \text{primaryInstrumentType}^{12}$$

Notice that we are now able to assert that Stradivarius_1707 is a Violin^1, while in the original MO this was not possible. We do not yet see any added value in terms of logical consequence though. Consider that our data set is combined with yet another one, describing a different Nigel Kennedy – a drummer. Assume that we have employed an entity recognition tool, which unified both individuals because of the same name. Now, such tools are probably unaware that an artist with two primary instruments of a very different nature is unlikely and should not be unified. However, a slightly modified version of MO, shown in the next example, allows us to detect this discrepancy.

Example 2. As instrument types are now concepts, we may model subsumption and disjointness between them properly with GCI axioms. In addition we assert functionality on the $\text{primaryInstrumentType}^{12}$ property:

$$\text{Violin}^1 \sqsubseteq \text{Strings}^1 \qquad \text{Drums}^1 \sqsubseteq \text{Percussions}^1 \qquad \text{Strings}^1 \sqsubseteq \neg\text{Percussions}^1$$
$$\text{MusicArtist}^1 \sqsubseteq {\leqslant}1\,\text{primaryInstrumentType}^{12}.\top^2$$

$$\text{NigelKennedy}: \text{MusicArtist}^1$$
$$\text{NigelKennedy}, \text{Violin}^1: \text{primaryInstrumentType}^{12}$$
$$\text{NigelKennedy}, \text{Drums}^1: \text{primaryInstrumentType}^{12}$$

The KB above is consistent, but the concepts Violin^1 and Drums^1 are both unsatisfiable: Due to functionality of $\text{primaryInstrumentType}^{12}$, they have the same intension. Thanks to intensional regularity (cf. Sect. 3.2), this implies they share the same extension, which must be empty, since their respective superconcepts are mutually disjoint. That is, the two concepts are inconsistent, an indicator that something went wrong.

Note that this example especially demonstrates that useful consequences are derived by metamodelling combined with reasoning about concepts, which is not possible when concepts are modelled by individuals.

4.2 Beyond the Second Order

While our logic can handle concepts of any finite order, only a small number of orders is typically needed. Pan and Horrocks [18] already noted that the object level (order 0) and three additional orders mostly suffice. To see a scenario that goes beyond the second order, let us consider the musical instrument types in a more complex fashion.

While generic notions of instruments, such as Violin or Drums, are 1st-order concepts, in a typical musical instrument classification, these concepts are further classified into 2nd-order instrument categories. What is more interesting, several instrument classifications have been developed. The classical classification sorts instruments into Woodwind, String, Percussion, and Electronic instruments (and subcatergories). Hornbostel and Sachs [11] classification sorts instruments into Idiophones, Membranophones,

Chordophones, Aerophones and Electrophones (with many more subcategories). However, some modern instruments do not fit in any of these types. The modern classification [14] thus extends Hornbostel-Sachs with additional categories. All instrument categories introduced by classifications are 2nd-order concepts (e.g., Violin is an instance of both String Instrument and Chordophone).

We can clearly see, that there are at least three different classifications. To make some order in the vast number of categories, we will introduce concepts Classical Instrument Category, Hornbostel-Sachs Instrument Category, Modern Instrument Category, etc. (all of order 3). Some interesting ontological relations between these 3rd-order concepts are apparent: Hornbostel-Sachs Instrument Category is a subconcept of the Modern Instrument Category, while Modern and Classical Instrument Category are disjoint.

More examples going up to the 3rd-order are:

- Our analysis of Fig. 1(b) uses three orders of concepts: an individual CD (order 0) is classified into its respective release (order 1), releases are classified into different types, such as Single, EP, Album (order 2), instances of Release Type (order 3).
- First-order concepts such as Professor, Violinist, Musicologist are rôles assumed by people, in this case, professional rôles. There are also other kinds of rôles, e.g., Supervisor and Apprentice (academic rôles), Father and Mother (family rôles). It makes sense to introduce 2nd-order concepts Professional Rôle, Academic Rôle, Family Rôle, which can be all classified into Human Rôle Type (order 3). Note that there can also be Animal Rôle Type (order 3), instantiated by, e.g., Animal Professional Rôle (order 2), classifying concepts such as Rescue Dog or Guard Dog (order 1).
- Most cars produced fall under a certain car model, e.g., VW Beetle or Citroën 2CW which are order 1 concepts, subconcepts of Passenger Car (order 1). The concept Car Model is of order 2. Car models are commonly classified as Small Family Car, Sports Car, SUV, . . . (all order 2), which are instances of the Car Model Class concept (order 3), while the concept Car Model is not its instance.

4.3 Ontological Background Models

Modelling of concepts as individuals and similar approximation techniques are manageable if used consistently in one ontology. However, in ontology integration and transformation, matching entities modelled at different levels of abstraction poses a problem. We have lately proposed so called PURO background modelling language [22] that enables to explicitly mark these levels of abstraction in ontologies by annotation. Consider one LD dataset that contains records relating artists to individual instruments, such as Stradivarius_1707, and another dataset relating artists to more abstract instrument types, such as mo-mit:Violin. When populating the latter with data from the former, we want to be aware of the difference, and make the necessary changes. This can be assisted by creating an *ontological background model* in which the levels of abstractions (i.e., orders) are made explicit. The $\mathcal{TH}(\mathcal{SROIQ})$ language may be further useful here, as it allows, in addition, to represent background models logically, reason about them, and represent also relations between the related entities at different levels of abstraction, e.g., by asserting all particular violins on one side as instances of the violin concept on the other side.

4.4 Relationship with Materialization

Materialization, as introduced by Pirotte at al. [20], is a relation between a more abstract class *A* and a more concrete class *C*, which is useful in conceptual modelling. The standard example provided by Pirotte et al. is when *A* is Car Model and *C* is Car. The "result" of materialization is that *C* is partitioned into a set of mutually disjoint subclasses (in this case, of particular car models) which group those members of *C* which are related to each respective member of *A*.

Interestingly, Pirotte at al. base their study of materialization on seeing the instances of *A*, e.g., car models, as "two-faceted" entities with an object facet and a class facet. The object facet describes the common features of a given model. The class facet has individual cars of the model as its instances. In our study, we have considered several such entities, e.g., music releases, musical instrument types, and also car models. In $\mathcal{TH}(\mathcal{SROIQ})$, they can be modelled as concepts of order $t > 0$, whose intensions $\cdot^{\mathcal{I}}$ correspond to the object facets, and extensions $\cdot^{\mathcal{E}}$ to the class facets. "Two-facetedness" of entities is an integral feature of our logic, which thus provides concepts with the ability to participate in roles and as instances in higher-order concepts.

This shows that, at least in some cases, materialization manipulates with higher-order constructs – the class *C* can often be viewed as one order up from *A*. Notably, this is not always the case: Pirotte et al. give, e.g., an example of stories materializing as plays, which, in turn, materialize as theatrical performances, where all involved classes are naturally of the first order.

Finally, while materialization is related to our work in the sense that we propose a DL language offering features which can capture some of the constructs involved (such as two-faceted entities, and the possibility to assert the meta-relation materializesAs between two classes) Pirotte et al. focus on inheritance of properties along the materialization relation, which we do not study in the current work. It is also obvious that our language enables to model relations between classes unrelated to materialization (modelling the different classifications of musical instrument types and relations between these classifications is a good example of that).

5 Related Work

In Table 4 we compare $\mathcal{TH}(\mathcal{SROIQ})$ to other similar proposals. Motik [15] and De Giacomo et al. [3] proposed higher-order DLs. Both employ HiLog-style semantics, they are untyped, and with heterogeneous higher-order entities (e.g., a concept may contain any other entity including itself; same entity can act as individual, concept, and role). This limits the expressivity of the underlying language (while maintaining decidability) to $\mathcal{ALCHOIQ}$ and $\mathcal{SHIQ}$, respectively. De Giacomo et al. [3] feature intensions also for complex expressions (roles, concepts).

To obtain decidability of $\mathcal{TH}(\mathcal{SROIQ})$, we have adapted a reduction originally proposed by Glimm et al. [4]. Their work is not primarily concerned with construction and properties of a higher-order DL, instead it offers a way to metamodel higher-order entities directly in $\mathcal{SROIQ}$. This methodology can be indeed used to capture a number of our intended applications. The main difference is that only second-order concepts are supported, and the approach is not characterized in model-theoretic terms.

Table 4. Comparison of related higher-order DLs

	Unlim. high ords.	Separat. ords.	Inter-ord. roles	Intensional regularity	Complex intensions	Extensio- nality	Expressivity
Punning	Y	N	Y	N	N	N	$\mathcal{SROIQ}$
OWL FA	Y	Y	N	Y	N	Y	$\mathcal{SHOIQ}$
Motik (v-sem.)	Y	N	Y	Y	N	N	$\mathcal{ALCHOIQ}$
De Giacomo et al.	Y	N	Y	Y	Y	N	$\mathcal{SHIQ}$
Glimm et al.	N	Y	Y (1–2)	Y	N	N	$\mathcal{SROIQ}$
$\mathcal{TH}(\mathcal{SROIQ})$	Y	Y	Y	Y	N	N	$\mathcal{SROIQ}$

In OWL 2, *punning* allows to use the same identifier as an individual and also as a class [5]. The two semantic denotations are completely independent, thus intensional regularity is violated. Hence punning is not very well suited for metamodelling [9,15].

Pan et al. proposed RDFS(FA)/OWL FA [18,19], a metamodelling architecture for the Semantic Web. Their approach is interesting as it also clearly separates the higher orders. This work uses Henkin semantics [8]; we have argued that due to its extensionality it may be too strong for metamodelling [9]. Also, roles connecting different orders are not allowed.

Works of Colucci et al. [2] and Szałas [23] apply second-order DLs to non-standard and common-sense reasoning tasks, respectively.

6 Conclusions and Future Work

We have proposed a $\mathcal{TH}(\mathcal{SROIQ})$, a higher-order variant of $\mathcal{SROIQ}$, with the distinctive feature of strictly separated hierarchy of types. Such a variant allows to overcome the first-order nature of OWL, and there is no additional cost for this, because, as we have showed, it is polynomially reducible back to regular $\mathcal{SROIQ}$, hence decidable, and with the same computational complexity (i.e., it is 2-NExpTime complete).

An arbitrary number of orders is probably not needed very often, but as we have showed, in certain metamodelling scenarios a combination of meta-concepts and meta-meta-concepts may be useful. $\mathcal{TH}(\mathcal{SROIQ})$ can also be used to directly express PURO background models [22] of existing ontologies and LD vocabularies, which is useful for analysis, debugging and integration purposes.

While the strict separation of types provides for some order, there are also some cases in which it is limiting, especially in "true" metamodelling, e.g., when capturing deprecation or authorship of ontology entities, one might want to have just one Deprecated meta-concept and definedBy meta-role to do this, regardless of the entity being marked. In the future we want to investigate a further extension of the language by such "promiscuous" concepts and roles.

References

1. Chen, W., Kifer, M., Warren, D.S.: A foundation for higher-order logic programming. J. Logic Programming 15(3), 187–230 (1993)
2. Colucci, S., Di Noia, T., Di Sciascio, E., Donini, F.M., Ragone, A.: Second-order description logics: Semantics, motivation, and a calculus. In: DL (2010)
3. De Giacomo, G., Lenzerini, M., Rosati, R.: On higher-order description logics. In: DL (2009)
4. Glimm, B., Rudolph, S., Völker, J.: Integrated metamodeling and diagnosis in OWL 2. In: ISWC (2010)
5. Golbreich, C., Wallace, E.K. (eds.): OWL 2 Web Ontology Language New Features and Rationale. Recommendation, W3C (27 October 2009)
6. Guarino, N., Welty, C.: An overview of ontoclean. In: Staab, S., Studer, R. (eds.) The Handbook on Ontologies, pp. 151–172. IHIS, Springer (2004)
7. Hayes, P. (ed.): RDF Semantics. Recommendation, W3C (10 February 2004)
8. Henkin, L.: Completeness in the theory of types. J. Symbolic Logic 15, 81–91 (1950)
9. Homola, M., Kľuka, J., Svátek, V., Vacura, M.: Towards typed higher-order description logics. In: DL (2013)
10. Homola, M., Kľuka, J., Svátek, V., Vacura, M.: Typed higher-order variant of sroiq (extended version). Tech. Rep. TR-2014-0042, Comenius University (2014), available online: `http://kedrigern.dcs.fmph.uniba.sk/reports/download.php?id=57`
11. von Hornbostel, E.M., Sachs, C.: Systematik der Musikinstrumente. Zeitschrift für Ethnologie 14, 553–590 (1914)
12. Horrocks, I., Kutz, O., Sattler, U.: The even more irresistible $\mathcal{SROIQ}$. In: KR (2006)
13. Kazakov, Y.: RIQ and SROIQ are harder than SHOIQ. In: KR. pp. 274–284 (2008)
14. Mann, S.: Natural interfaces for musical expression: Physiphones and a physics-based organology. In: Proceedings of the International Conference on New Interfaces for Musical Expression. pp. 118–123 (2007)
15. Motik, B.: On the properties of metamodeling in OWL. J. Log. Comput 17(4), 617–637 (2007)
16. Noy, N. (ed.): Representing Classes As Property Values on the Semantic Web. Working Group Note, W3C (5 April 2005)
17. OWL Working Group (ed.): OWL 2 Web Ontology Language Document Overview. Recommendation, W3C (27 October 2009)
18. Pan, J.Z., Horrocks, I.: RDFS(FA): Connecting RDF(S) and OWL DL. IEEE Trans. Knowl. Data Eng. 19, 192–206 (2007)
19. Pan, J.Z., Horrocks, I., Schreiber, G.: Owl fa: A metamodeling extension of owl dl. In: OWLED (2005)
20. Pirotte, A., Zimányi, E., Massart, D., Yakusheva, T.: Materialization: A powerful and ubiquitous abstraction pattern. In: VLDB. pp. 630–641. Morgan Kaufmann (1994)
21. Raimond, Y., Giasson, F. (eds.): Music Ontology Specification (28 November 2010), available online: `http://purl.org/ontology/mo/`
22. Svátek, V., Homola, M., Kľuka, J., Vacura, M.: Metamodeling-based coherence checking of OWL vocabulary background models. In: OWLED (2013)
23. Szałas, A.: Second-order reasoning in description logics. Journal of Applied Non-Classical Logics 16(3-4), 517–530 (2006)