Contextualizing DL axioms: Formalization, a New Approach, and its Properties

Antoine Zimmermann[1] and José M. Giménez-García[2]

[1] Univ Lyon, MINES Saint-Étienne, CNRS, Laboratoire Hubert Curien
UMR 5516, F-42023 Saint-Étienne, France
`antoine.zimmermann@emse.fr`
[2] Université de Lyon, CNRS, UMR 5516,
Laboratoire Hubert-Curien, Saint-Étienne, France
`jose.gimenez.garcia@univ-st-etienne.fr`

Abstract. We address the problem of providing contextual information about a logical formula (*e.g.*, provenance, date of validity, or confidence) and representing it within a logical system. In this case, it is needed to rely on a higher order or non standard formalism, or some kind of reification mechanism. We explore the case of reification and formalize the concept of contextualizing logical statements in the case of Description Logics. Then, we define several properties of contextualization that are desirable. No previous approaches satisfy all of the them. Consequently, we define a new way of contextually annotating statements. It is inspired by NdFluents, which is itself an extension of the 4dFluents approach for annotating statements with temporal context. In NdFluents, instances that are involved in a contextual statement are sliced into contextual parts, such that only parts in the same context hold relations to one another, with the goal of better preserving inferences. We generalize this idea by defining contextual parts of relations and classes. This formal construction better satisfies the properties, although not entirely. We show that it is a particular case of a general mechanism that NdFluents also instantiates, and present other variations.

Keywords: Annotations, Contexts, Metadata, Ontologies, Description Logic, Semantic Web, Reasoning, Reification

1 Introduction

The problem of being able to reason not only *with* logical formulas, but also *about* said formulas, is an old one in artificial intelligence. McCarthy [1] proposed to extend first order logic by reifying context and formulas to introduce a binary predicate $\mathsf{ist}(\phi, c)$ satisfied if the formula ϕ is true (ist) in the context c. However, a complete axiomatization and calculus for McCarthy's contextual logic has never been formalized. Giunchiglia [2] proposed the grouping of "local" formulas in contexts, and then using other kinds of formulas to characterize how knowledge from multiple contexts is *compatible*. This idea of locality+compatibility [3] has led to several non standard formalisms for reasoning with multiple contexts [4].

Alternatively, the approach of annotated logic programming [5] considers that a contextual annotation is just a value in an algebraic structure (*e.g.*, a number or a temporal interval). This idea was later applied to annotated RDF and RDFS [6, 7].

The representation of statement annotation has sometimes being thought of as a data model problem without consideration of the logical formalism behind. In particular, several proposals to extend the RDF data model in various ways for allowing annotations have been made: named graphs [8], RDF+ [9], RDF* [10], Yago Model [11]. However, the underlying data structures have not a clear formal semantics. Therefore, some authors advocate another approach to representing annotation of knowledge: reify the statement or its context and describe it within the formalism of the statement. This requires modifying the statement so as to integrate knowledge of the context or statement. Examples of such techniques are reification [12], N-Ary Relations [13], Singleton Property [14]), and NdFluents [15]. *This paper provides an abstraction of the reification techniques in the context of Description Logics (DLs) in the form of what we call contextualization functions.* Additionally, we introduce a new technique for the representation of contextual annotations that satisfies better some desirable properties.

After introducing our notations for DLs in Sec. 2, we provide formal definitions that allow us to define verifiable properties of the reification techniques (Sec. 3). Our new technique, named *NdTerms*, is presented in Sec. 4, where we also prove to what extent it satisfies the properties of the previous section. Sec. 5 discuss some of the problems that may occur when combining knowledge having different annotations. In Sec. 6, we present how the other approaches fit in our formalization and why they do not satisfy well the properties. Finally, we discuss this and future work in Sec. 7.

2   Preliminaries

In this section, we introduce the notations and definitions we use in relation to Description Logics. Note that we use an extended version of DL where all terms can be used as *concept names*, *role names*, and *individual names* in the same ontology. Using the same name for different types of terms is known as "punning" in OWL 2 [16, Section 2.4.1]. Moreover, we allow more constructs than in OWL 2 DL and make no restriction on their use in order to show that our approach is not limited to a specific DL.

We assume that there is an infinite set of *terms* N. Every term is an *individual*. A *role* is either a term or, given roles R and S, $R \sqcup S$, $R \sqcap S$, $\neg R$, R^-, $R \circ S$ and R^+. A *concept* is either a term, or, given concepts C, D, role R, individuals $u_1, \ldots, u_k$, and natural number n, $\bot$, $\top$, $C \sqcup D$, $C \sqcap D$, $\exists R.C$, $\forall R.C$, $\leq nR.C$, $\geq nR.C$, $\neg C$ or $\{u_1, \ldots, u_k\}$. Finally, we also allow concept product $C \times D$ to define a role.

Interpretations are tuples $\langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}_u}, \cdot^{\mathcal{I}_r}, \cdot^{\mathcal{I}_c} \rangle$, where $\Delta^{\mathcal{I}}$ is a non-empty set (the domain of interpretation) and $\cdot^{\mathcal{I}_u}$, $\cdot^{\mathcal{I}_r}$, and $\cdot^{\mathcal{I}_c}$ are the interpretation functions for individuals, roles and concepts respectively such that:

- for all $u \in \mathsf{N}$, $u^{\mathcal{I}_u} \in \Delta^{\mathcal{I}}$;
- for all $P \in \mathsf{N}$, $P^{\mathcal{I}_r} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ and interpretation of roles is inductively defined by $(R \sqcup S)^{\mathcal{I}_r} = R^{\mathcal{I}_r} \cup S^{\mathcal{I}_r}$, $(R \sqcap S)^{\mathcal{I}_r} = R^{\mathcal{I}_r} \cap S^{\mathcal{I}_r}$, $(\neg R)^{\mathcal{I}_r} = (\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}) \backslash R^{\mathcal{I}_r}$, $(R^-)^{\mathcal{I}_r} = \{\langle x, y \rangle | \langle y, x \rangle \in R^{\mathcal{I}_r}\}$, $(R \circ S)^{\mathcal{I}_r} = \{\langle x, y \rangle | \exists z.\langle x, z \rangle \in R^{\mathcal{I}_r} \wedge \langle z, y \rangle \in S^{\mathcal{I}_r}\}$ and $(R^+)^{\mathcal{I}_r}$ is the reflexive-transitive closure of $R^{\mathcal{I}_r}$ (with R and S being arbitrary roles).
- for all $A \in \mathsf{N}$, $A^{\mathcal{I}_c} \subseteq \Delta^{\mathcal{I}}$ and interpretation of concepts is defined by $\bot^{\mathcal{I}_c} = \emptyset$, $\top^{\mathcal{I}_c} = \Delta^{\mathcal{I}}$, $(C \sqcup D)^{\mathcal{I}_c} = C^{\mathcal{I}_c} \cup D^{\mathcal{I}_c}$, $(C \sqcap D)^{\mathcal{I}_c} = C^{\mathcal{I}_c} \cap D^{\mathcal{I}_c}$, $(\exists R.C)^{\mathcal{I}_c} = \{x | \exists y. y \in C^{\mathcal{I}_c} \wedge \langle x, y \rangle \in R^{\mathcal{I}_r}\}$, $(\forall R.C)^{\mathcal{I}_c} = \{x | \forall y. \langle x, y \rangle \in R^{\mathcal{I}_r} \Rightarrow y \in C^{\mathcal{I}_c}\}$, $(\leq nR.C)^{\mathcal{I}_c} = \{x | \sharp\{y \in C^{\mathcal{I}_c} | \langle x, y \rangle \in R^{\mathcal{I}_r}\} \leq n\}$, $(\geq nR.C)^{\mathcal{I}_c} = \{x | \sharp\{y \in C^{\mathcal{I}_c} | \langle x, y \rangle \in R^{\mathcal{I}_r}\} \geq n\}$, $(\neg C)^{\mathcal{I}_c} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}_c}$, $\{u_1, \ldots, u_k\} = \{u_1^{\mathcal{I}_u}, \ldots, u_k^{\mathcal{I}_u}\}$, where C and D are arbitrary concepts, R an arbitrary role, $u_1, \ldots, u_k$ are individual names, and k and n two natural numbers.
- Roles defined as a concept product are interpreted as $(C \times D)^{\mathcal{I}_r} = C^{\mathcal{I}_c} \times D^{\mathcal{I}_c}$ for arbitrary concepts C and D.

In the following, we slightly abuse notations by defining interpretations as pairs $\langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ where $\cdot^{\mathcal{I}}$ denotes the three functions $\cdot^{\mathcal{I}_u}$, $\cdot^{\mathcal{I}_c}$, and $\cdot^{\mathcal{I}_r}$. Moreover, when we write $x^{\mathcal{I}} = y^{\mathcal{I}'}$, it means "$x^{\mathcal{I}_u} = y^{\mathcal{I}'_u}$ and $x^{\mathcal{I}_c} = y^{\mathcal{I}'_c}$ and $x^{\mathcal{I}_r} = y^{\mathcal{I}'_r}$".

Axioms are either general concept inclusions $C \sqsubseteq_c D$, sub-role axioms $R \sqsubseteq_r S$, instance assertions $C(a)$, or role assertions $R(a, b)$, where C and D are concepts, R and S are roles, and a and b are individual names. An interpretation $\mathcal{I}$ *satisfies* axiom $C \sqsubseteq_c D$ iff $C^{\mathcal{I}_c} \subseteq D^{\mathcal{I}_c}$; it satisfies $R \sqsubseteq_r S$ iff $R^{\mathcal{I}_r} \subseteq S^{\mathcal{I}_r}$; it satisfies $C(a)$ iff $a^{\mathcal{I}_u} \in C^{\mathcal{I}_c}$; and it satisfies $R(a, b)$ iff $\langle a^{\mathcal{I}_u}, b^{\mathcal{I}_u} \rangle \in R^{\mathcal{I}_r}$. When $\mathcal{I}$ satisfies an axiom α, it is denoted by $\mathcal{I} \models \alpha$. Instance assertions, role assertions and individual identities constitute the ABox axioms.

An ontology O is composed of a set of terms called the signature of O and denoted by $\mathsf{Sig}(O)$, and a set of axioms denoted by $\mathsf{Ax}(O)$. An interpretation $\mathcal{I}$ is a model of an ontology O iff for all $\alpha \in \mathsf{Ax}(O)$, $\mathcal{I} \models \alpha$. In this case, we write $\mathcal{I} \models O$. The set of all models of an ontology O is denoted by $\mathrm{Mod}(O)$. A *semantic consequence* of an ontology O is a formula α such that for all $\mathcal{I} \in \mathrm{Mod}(O), \mathcal{I} \models \alpha$.

In the rest of the paper, we will use `teletype font` to denote known individuals, and normal font for unknown individuals and variables (*e.g.*, `City`(`babylon`) and `City`(x)).

3 Contextualization of Statements

A contextual annotation can be thought of as a set of ABox axioms that describe an individual representing the statement (the anchor) that is annotated. An annotated statement (or ontology) is the combination of a DL axiom (or DL ontology) with a contextual annotation.

Definition 1 (Connected individuals). *Two terms a and b are* connected individuals *wrt an ABox A iff a and b are used as individual names in A, and either*

- *a and b are the same term, or*

– *there exists $R_1, \cdots, R_n$ and $z_1, \cdots, z_{n-1}$, such that:*
  - $R_1(a, z_1)$, *or* $R_1(z_1, a)$
  - $R_i(z_{i-1}, z_i)$, *or* $R_i(z_i, z_{i-1})$, $2 \leq i \leq n-2$
  - $R_n(z_{n-1}, b)$, *or* $R_n(b, z_{n-1})$

Example 1. If we consider the ABox $A = \{P(a, b), Q(c, b), S(d, e)\}$, the pairs of individuals $\{a, b\}$, $\{b, c\}$, $\{a, c\}$, and $\{d, e\}$ are *connected individuals*, but $\{a, d\}$, $\{b, d\}$, $\{c, d\}$, $\{a, e\}$, $\{b, e\}$, and $\{c, e\}$ are not.

Definition 2 (Contextual annotation). *A* contextual annotation *Ca is an ABox with signature* $\{\mathbf{a}\} \cup \Sigma$ *where* $\mathbf{a} \notin \Sigma$ *is a distinguished term (called the* anchor*) and Σ is a DL signature such that* $\forall x \in \Sigma$, $\{a, x\}$ *are* connected *individuals.*

Example 2. The Abox $CA = \{\texttt{validity}(\mathbf{a}, \mathbf{t}), \texttt{Interval}(\mathbf{t}) \texttt{ from}(\mathbf{t}, 609\text{BC}), \texttt{to}(\mathbf{t}, 539\text{BC}), \texttt{prov}(\mathbf{a}, \mathbf{w}), \texttt{name}(\mathbf{w}, \texttt{wikipedia}), \texttt{Wiki}(\mathbf{w})\}$ is a *contextual annotation*, where $\mathbf{a}$ is the anchor and $\Sigma = \{\mathbf{t}, \texttt{Interval}, \mathbf{w}, \texttt{wikipedia}, \texttt{wiki}, 609\text{BC}, 539\text{BC}\}$.

Definition 3 (Annotated statement). *An* annotated statement *is a pair $\langle \alpha, Ca \rangle$ such that α is a description logic axiom and Ca is a contextual annotation.*

Example 3. The pair $\langle \alpha, CA \rangle$, where $\alpha = \texttt{capital}(\texttt{babylon}, \texttt{babylonianEmpire})$ and CA is the contextual annotation from Ex. 2, is an *annotated statement*.

Definition 4 (Annotated ontology). *An* annotated ontology *is a pair $\langle O, Ca \rangle$ such that O is a description logic ontology and Ca is a contextual annotation.*

Each reification technique has an implicit construction plan in order to map an annotated statement to a resulting ontology. A contextualization (Def. 5) represents the procedure that generates a single DL ontology from a given annotated statement or ontology. The procedure must not lose information, especially not the annotation.

Definition 5 (Contextualization). *A* contextualization *is a function f that maps each annotated statement $\alpha_{Ca} = \langle \alpha, Ca \rangle$ to a description logic ontology $f(\alpha_{Ca}) = \mathsf{St}(\alpha_{Ca}) \cup \mathsf{Cx}(\alpha_{Ca})$ such that:*

– *there exists an individual u in the signature of $\mathsf{St}(\alpha_{Ca})$ and of $\mathsf{Cx}(\alpha_{Ca})$ such that:*
  - *for all $R(\mathbf{a}, x) \in Ca$, $R(u, x) \in \mathsf{Cx}(\alpha_{Ca})$;*
  - *for all $R(x, \mathbf{a}) \in Ca$, $R(x, u) \in \mathsf{Cx}(\alpha_{Ca})$;*
  - *for all $C(\mathbf{a}) \in Ca$, $C(u) \in \mathsf{Cx}(\alpha_{Ca})$;*
  - *for all other $\alpha \in Ca$, $\alpha \in \mathsf{Cx}(\alpha_{Ca})$.*
– *there is an injective mapping between the signature of α and the signature of $\mathsf{St}(\alpha_C)$.*

We extend f to all annotated ontologies $O_{Ca} = \langle O, Ca \rangle$ by defining $f(O_{Ca}) = \bigcup_{\alpha \in O} f(\langle \alpha, Ca \rangle)$.

Example 4. An example contextualization function f_{ex} introduces a fresh term t for each annotated statement with a role assertion $R(a, b)$, where R, a, and b are three therms, creates new axioms $\mathtt{subject}(t, s)$, $\mathtt{predicate}(t, R)$, $\mathtt{object}(t, o)$, and finally removes the axiom $R(a, b)$. Notice that this construction requires the punning of term R. This function is analogous to *RDF Reification*. The result of this contextualization, along with Other possible known approaches, is described in Sec. 6.

Those are the only structures that we will consider in this paper. The remaining definitions are desirable properties that a contextualization should satisfy, especially if one wants it to preserve as much of the original knowledge as possible.

Definition 6 (Soundness). *A contextualization function f is* sound *wrt a set of annotated ontologies Ω iff for each $O_{Ca} = \langle O, Ca \rangle \in \Omega$ such that O and Ca are consistent, then $f(O_{Ca})$ is consistent.*

That is, a contextualization is sound if, when contextualizing a consistent ontology, the result is also consistent. This property avoids that the contextualization introduces unnecessary contradictions that would result in everything being entailed by it. Note that this requirement is not necessary in the opposite direction, *i.e.*, if $f(O_{Ca})$ is consistent, it is not required that O and C are consistent.

Example 5. The contextualization function f_{ex} from Ex. 4 is sound wrt the set of ontologies Ω, where $\Omega \cup \{\mathtt{subject}, \mathtt{predicate}, \mathtt{predicate}\} = \emptyset$.

Definition 7 (Inconsistency preservation). *Let f be a contextualization function. We say that f* preserves inconsistencies *iff for all annotated ontologies $O_{Ca} = \langle O, Ca \rangle$, if O is inconsistent then $f(O_{Ca})$ is inconsistent.*

Inconsistency preservation means that a self-contradictory ontology in a given context is contextualized into an inconsistent ontology, such that bringing additional knowledge from other contexts would result in no more consistency. If something is inconsistent within a context, then it is not really worth to consider reasoning with this annotated ontology.

Example 6. The contextualization function f_{ex} from Ex. 4 does not preserve inconsistencies. For instance, $\mathtt{capitalOf}$ can be defined as irreflexive using the following axiom: $\exists\mathtt{capitalOf}.\top \sqsubseteq \forall\mathtt{capitalOf}^-.\bot$. Then, the axiom $\mathtt{capitalOf(babylon, babylon)}$ would make the ontology inconsistent. But when applying f_{ex} the result is consistent.

Definition 8 (Entailment preservation). *Let f be a contextualization function. Given two description logic ontologies O_1 and O_2 such that $O_1 \models O_2$, we say that f* preserves *the entailment* between O_1 and O_2 *iff for all contextual annotations Ca, $f(\langle O_1, Ca \rangle) \models f(\langle O_2, Ca \rangle)$. Given a set $\mathfrak{K} \subseteq \mathsf{N}^a$ of contextual annotations, if f preserves all entailments between ontologies in $\mathfrak{K}$, then we say that f is* entailment preserving *for $\mathfrak{K}$.*

In short, a contextualization is entailment preserving if all the knowledge that could be inferred from the original ontology can also be inferred, in the same context, in the contextualized ontology.

Example 7. The contextualization function f_{ex} from Ex. 4 preserves entailments for the TBox of ontologies (because no modifications are made on its axioms), but it does not preserve entailments on role assertions. For instance, the axioms `capitalOf` $\sqsubseteq$ `cityOf`, `capitalOf(babylon, babylonianEmpire)` entails `cityOf(babylon, babylonianEmpire)`, but this inference is not preserved after applying f_{ex}.

4 The *NdTerms* Approach

This section defines the *NdTerms* approach that extends the NdFluent proposal [15]. To this end, we assume that terms are divided into three infinite disjoint sets $\mathsf{N^{nc}}$, $\mathsf{N^c}$, and $\mathsf{N^a}$ called the *non contextual terms*, the *contextual terms*, and the *anchor terms* respectively. We also assume that there is an injective function $\mathfrak{A} : \mathfrak{C} \to \mathsf{N^a}$ and for all $Ca \in \mathfrak{C}$ there is an injective function $\mathsf{ren}_{Ca} : \mathsf{N^{nc}} \to \mathsf{N^c}$ and two terms $\mathtt{isContextualPartOf}, \mathtt{isInContext} \in \mathsf{N^{nc}}$. For any Ca, we extend ren_{Ca} to axioms by defining $\mathsf{ren}_{Ca}(\alpha)$ as the axiom built from α by replacing all terms $t \in \mathsf{Sig}(\alpha)$ with $\mathsf{ren}_{Ca}(t)$.

4.1 Contextualization Function in *NdTerms*

The contextualization needs to combine the ontologies from the statements and the contextual annotation. However, if we naïvely make the union of the axioms, they could contradict, and it would not be possible to ensure the desired properties. For example, an ontology may restrict the size of the domain of interpretation to be of a fixed cardinality, while the contextual annotation may rely on more elements outside the local universe of this context. For this reason we use the concept or relativization: The ontology is modified in such a way that the interpretation of everything explicitly described in it is confined to a set, while external terms or constructs may have elements outside said set. Relativization has been applied in various logical settings over the past four decades (*e.g.*, [17]) and applied to DLs and OWL [18], among others.

Definition 9 (Relativization). *Let Ca be a contextual annotation. Given an ontology O, the relativization of O in Ca is an ontology $\mathsf{Rel}_{Ca}(O)$ built from O as follows:*

1. $\mathsf{Sig}(\mathsf{Rel}_{Ca}(O)) = \mathsf{Sig}(O) \cup \top_{Ca}$ *where $\top_{Ca}$ is a term not appearing in $\mathsf{Sig}(O)$;*
2. *for all appearances of $\top$ in an axiom of O, replace $\top$ with $\top_{Ca}$;*
3. *for all concepts $\neg C$ appearing in an axiom of O, replace it with $\neg C \sqcap \top_{Ca}$;*
4. *for all roles $\neg R$ appearing in an axiom of O, replace it with $\neg R \sqcap (\top_{Ca} \times \top_{Ca})$;*
5. *for all concepts $\forall R.C$ appearing in an axiom of O, replace it with $\forall R.C \sqcap \top_{Ca}$;*
6. *for all roles R^+ appearing in an axiom of O, replace it with $R^+ \sqcap \top_{Ca} \times \top_{Ca}$.*

7. *Additionally, for all terms $t \in \mathsf{Sig}(O)$, the following axioms are in $\mathsf{Rel}_{Ca}(O)$:*
    - $t \sqsubseteq \top_{Ca}$,
    - $\top_{Ca}(t)$,
    - $\exists t.\top \sqsubseteq \top_{Ca}$,
    - $\top \sqsubseteq \forall t.\top_{Ca}$.

The relativization of an ontology can be done systematically by relativizing its concepts and roles, which in turn can be achieved by using Def. 10.

Definition 10 (Relativization of concepts and roles). *Given a contextual annotation Ca, we define a function rel_{Ca} that maps concepts and roles to concepts and roles according to the rules of Items 2-6. That is, recursively:*

- $\mathsf{rel}_{Ca}(t) = t$;
- $\mathsf{rel}_{Ca}(\{u_1, \ldots u_k\}) = \{\mathsf{rel}_{Ca}(u_1), \ldots relf(u_k)\}$;
- $\mathsf{rel}_{Ca}(C \sqcup D) = \mathsf{rel}_{Ca}(C) \sqcup \mathsf{rel}_{Ca}(D)$;
- $\mathsf{rel}_{Ca}(C \sqcap D) = \mathsf{rel}_{Ca}(C) \sqcap \mathsf{rel}_{Ca}(D)$;
- $\mathsf{rel}_{Ca}(\neg C) = \neg\mathsf{rel}_{Ca}(C) \sqcap \top_{Ca}$;
- $\mathsf{rel}_{Ca}(C \times D) = \mathsf{rel}_{Ca}(C) \times \mathsf{rel}_{Ca}(D)$;
- $\mathsf{rel}_{Ca}(R \sqcup S) = \mathsf{rel}_{Ca}(R) \sqcup \mathsf{rel}_{Ca}(S)$;
- $\mathsf{rel}_{Ca}(R \sqcap S) = \mathsf{rel}_{Ca}(R) \sqcap \mathsf{rel}_{Ca}(S)$;
- $\mathsf{rel}_{Ca}(R \circ S) = \mathsf{rel}_{Ca}(R) \circ \mathsf{rel}_{Ca}(S)$;
- $\mathsf{rel}_{Ca}(\neg R) = \neg\mathsf{rel}_{Ca}(R) \sqcap \top_{Ca} \times \top_{Ca}$;
- $\mathsf{rel}_{Ca}(R^-) = \mathsf{rel}_{Ca}(R)^-$;
- $\mathsf{rel}_{Ca}(R^+) = \mathsf{rel}_{Ca}(R)^+ \sqcap \top_{Ca} \times \top_{Ca}$;
- $\mathsf{rel}_{Ca}(\exists R.C) = \exists\mathsf{rel}_{Ca}(R).\mathsf{rel}_{Ca}(C)$;
- $\mathsf{rel}_{Ca}(\forall R.C) = \forall\mathsf{rel}_{Ca}(R).\mathsf{rel}_{Ca}(C) \sqcap \top_{Ca}$;
- $\mathsf{rel}_{Ca}(\geq nR.C) = \geq n\ \mathsf{rel}_{Ca}(R).\mathsf{rel}_{Ca}(C)$;
- $\mathsf{rel}_{Ca}(\leq nR.C) = \leq n\ \mathsf{rel}_{Ca}(R).\mathsf{rel}_{Ca}(C)$.

where t is a term, C, D are concepts, R, S are roles, $u_1, \ldots u_k$ are individuals, and k, n are natural numbers.

Example 8. The axiom $\exists\mathtt{capitalOf}.\top \sqsubseteq \forall\mathtt{capitalOf}^-.\bot$ from Ex. 6 is relativized into $\exists\mathtt{capitalOf}.\top_{Ca} \sqsubseteq \forall\mathtt{capitalOf}^-.\bot \sqcap \top_{Ca}$.

Then, the contextualization in *NdTerms* is done by: (1) creating the replacement of the anchor using the function $\mathfrak{A}$, (2) renaming all the terms in the statement using the ren function, (3) linking them to the original terms by the $\mathtt{isContextualPartOf}$ relation, and (4) linking the renamed terms to the context using the $\mathtt{isInContext}$ relation.

Definition 11 (Contextualization in *NdTerms*). *Let $Ca \in \mathfrak{C}$ be any contextual annotation. Let $\alpha_{Ca} = \langle \alpha, Ca \rangle$ be an annotated statement such that the signatures of α and Ca are in $\mathsf{N^{nc}}$. We define the contextualization function f_{nd} such that $f_{\mathsf{nd}}(\alpha_{Ca}) = \mathsf{St}(\alpha_{Ca}) \cup \mathsf{Cx}(Ca)$ and:*

- $\mathsf{St}_{Ca}(\alpha) = \{\mathsf{ren}_{Ca}(\mathsf{Rel}_{Ca}(\alpha))\} \cup \{\mathtt{isContextualPartOf}(\mathsf{ren}_{Ca}(t), t) \mid t \in \mathsf{Sig}(\alpha)\} \cup \{\mathtt{isInContext}(\mathsf{ren}_{Ca}(t), \mathfrak{A}(Ca)) \mid t \in \mathsf{Sig}(\alpha)\}$.

– $\mathsf{Cx}(\mathit{Ca})$ *contains exactly the following axioms:*
  - *for all $R(\mathbf{a}, x) \in \mathit{Ca}$, $R(\mathfrak{A}(\mathit{Ca}), x) \in \mathsf{Cx}(\alpha)$;*
  - *for all $R(x, \mathbf{a}) \in \mathit{Ca}$, $R(x, \mathfrak{A}(\mathit{Ca})) \in \mathsf{Cx}(\mathit{Ca})$;*
  - *for all $C(\mathbf{a}) \in \mathit{Ca}$, $C(\mathfrak{A}(\mathit{Ca})) \in \mathsf{Cx}(\mathit{Ca})$;*
  - *for all other axioms $\beta \in \mathit{Ca}$, $\beta \in \mathsf{Cx}(\mathit{Ca})$.*

Similarly to Ex. 4, this construction requires punning, since all terms in the statement are used as individual names in the role assertion `isContextualPartOf`($\mathsf{ren}_{\mathit{Ca}}(t)$, t).

Example 9. The NdTerms contextualization of our running example within the context CA of Ex. 2 contains the following axioms, where $\mathit{term@Ca}$ is the result of the renaming function $\mathsf{ren}_{\mathit{Ca}}(\mathit{term})$:

> `capitalOf@`CA(`babylon@`CA, `babylonianEmpire@`CA)
> `isContextualPartOf`(`babylon@`CA, `babylon`)
> `isContextualPartOf`(`babylonianEmpire@`CA, `babylonianEmpire`)
> `isInContext`(`babylon`, `exampleContext`)
> `isInContext`(`babylonianEmpire`, `exampleContext`)
> `validity`(`exampleContext`, t) *etc.*

4.2 Properties of *NdTerms*

This subsection presents the properties satisfied by *NdTerms*. Due to space constraints, we omit the proofs of the theorems, but they are provided in the accompanying technical report [19].

The contextualization of *NdTerms* is sound, but only wrt annotated ontologies that satisfy certain conditions. In order to present the conditions, we need to introduce the following definition, that is also used in several proofs of this paper.

Definition 12 (Domain extensibility). *Let O be an ontology. A model $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ of O is* domain extensible *for O iff for all sets Δ^{+}, $\mathcal{I}' = \langle \Delta^{\mathcal{I}} \cup \Delta^{+}, \cdot^{\mathcal{I}} \rangle$ is also a model of O. An ontology is said to be* model extensible *iff it has a model that is domain extensible.*

Note that, even if the domain of interpretation of an ontology is infinite, that does not necessarily mean that its models are domain extensible. This notion is closely related to the notion of *expansion* in [18] since if $\mathcal{I}$ is domain extensible, then one can build infinitely many expansions of it.

Theorem 1 (Soundness of *NdTerms*). *Let Ca be a contextual annotation. If Ca has its signature in N^{nc} and is model extensible, then the contextualization function f_{nd} is sound wrt annotated ontologies $O_{\mathit{Ca}} = \langle O, \mathit{Ca} \rangle$, $\mathsf{Sig}(O) \subseteq \mathsf{N}^{\mathsf{nc}}$, and $\mathsf{Sig}(O) \cap \mathsf{Sig}(\mathit{Ca}) = \emptyset$.*

A model of the union of the of two ontologies can be made from the union of the models of the original ontologies if both ontologies are domain extensible. However, this is not a strong restriction, because the relativization of any consistent ontology is model extensible. Since *NdTerms* relativizes the ontology O and requires the contextual annotation Ca to be model extensible, it follows that a model can be made. Hence *NdTerms* is sound.

Theorem 2 (Inconsistency preservation of *NdTerms*). *The contextualization function f_{nd} preserves inconsistencies.*

If the contextualization of an ontology $f_{\mathsf{nd}}(O_{Ca})$ is consistent, then the original ontology O is necessarily consistent. Because the renaming function ren_{Ca} is injective, there exists an inverse function $\mathsf{ren}^{-}_{Ca} : \mathsf{ren}_{Ca}(\mathsf{N}^{\mathsf{nc}}) \to \mathsf{N}^{\mathsf{nc}}$ which is itself injective, and the renaming of terms gives us a model. The relativization function rel_{Ca}, on the other hand, does not change the interpretation (since $\top^{\mathcal{I}''} = \top^{\mathcal{I}''}_{Ca}$, adding $\sqcap \top_{Ca}$ to a concept does not change its interpretation, and replacing $\top$ with $\top_{Ca}$ has no effect on the interpretation of the concepts or roles). Therefore, if a model satisfies the contextualization function $f_{\mathsf{nd}}(O_{Ca})$, it also satisfies O, and it is not possible to have a consistent *NdTerms* contextualization of a non-consistent ontology.

In [15], we were only able to study entailment preservation in the limited setting of $pD*$ entailment. Here we present a much stronger theorem for *NdTerms*.

Theorem 3 (Entailment preservation of *NdTerms*). *Let Ca be a contextual annotation, let Ω be a set of ontologies having their signatures in N^{nc} and disjoint from the signature of Ca. If Ca is model extensible, then NdTerms is entailment preserving for $\{\langle O, Ca \rangle\}_{O \in \Omega}$.*

If the ontology O is inconsistent, the entailment is trivially preserved. If not, considering that the relativization of a consistent ontology preserves entailments, and that the function ren_{Ca} is just a renaming (and "truth is invariant under change of notation" [20]), the entailments of the ontology O are preserved. Moreover, all axioms in $\mathsf{Cx}(Ca)$ are satisfied by the interpretation $\mathcal{I}$, and whenever $\mathtt{isContextualPartOf}(\mathsf{ren}_{Ka}, t)$ or $\mathtt{isInContext}(\mathsf{ren}_{Ka}(t), \mathfrak{A}(Ka))$ are in the entailed contextualized ontology, then they are in original as well as well. Therefore, all entailments are preserved, and *NdTerms* satisfies this property.

5 Annotations in Multiple Contexts

So far, we assumed that all axioms of an ontology are annotated with the same contextual information. In this setting, the core of the contextualization function in *NdTerms* amounts to relativizing the axioms and renaming the terms. The renaming part may seem surprising because, as we said a couple of times already, "truth is invariant under change of notation". However, the usefulness of the renaming part becomes apparent when we want to combine several annotated ontologies having different contextual annotations (say Ca_1 and Ca_2). In this

case, if the renaming functions ren_{Ca_1} and ren_{Ca_2} are mapping non contextual terms into disjoint sets of contextual terms, then the contextualization function f_{nd} ensures that any inference made in a context will not interact with the knowledge from another context. This avoids the contextualized knowledge to be inconsistent when combining statements in different contexts that contradict each others.

The properties presented in Sec. 3 require a little adaptation when applied to the multi-contextual setting. Indeed, in spite of the soundness theorem of Sec. 4.2, in the general case if a set of annotated ontologies $\{\langle O_i, Ca_i\rangle\}_{i\in I}$ are satisfying the constraints of Theorem 1, it is still possible that $\bigcup_{i\in I} f_{\mathsf{nd}}(\langle O_i, Ca_i\rangle)$ is inconsistent. We expect that the preservation of consistency can be guaranteed if *all* the signatures of $\{O_i\}$ are disjoint from *all* the signatures of $\{Ca_i\}$. Studying in more details the case of multiple contextual annotations is planned for future work.

6 Other Approaches

Here we briefly present the most relevant reification approaches in the Semantic Web. For all of them, the contextualization only annotates the role assertions, leaving other axioms unmodified.

As seen in Ex. 4, *RDF reification* replaces $\alpha = R(x, y)$ with three new role assertions $\mathtt{subject}(\mathbf{a}_\alpha^{Ca}, x)$, $\mathtt{predicate}(\mathbf{a}_\alpha^{Ca}, R)$, and $\mathtt{object}(\mathbf{a}_\alpha^{Ca}, y)$ and the axioms in the contextual annotation are anchored on $\mathbf{a}_\alpha^{Ca}$, which depends on the role assertion R and the contextual annotation Ca. As shown in Ex. 6 and 7, this contextualization method preserves neither inconsistencies (in the sense of Def. 7) nor entailments on role assertions.

Example 10. An *RDF reification* contextualization of our running example within the context CA of Ex. 2 contains the following axioms:
    $\mathtt{subject(stbcobe, babylon)}$
    $\mathtt{predicate(stbcobe, capital)}$
    $\mathtt{object(stbcobe, babylonianEmpire)}$
    $\mathtt{validity(stbcobe, t)}$
    etc.

N-Ary relations replaces $R(x, y)$ by two role assertions $p_1(R)(x, \mathbf{a}_\alpha^{Ca})$ and $p_2(R)(\mathbf{a}_\alpha^{Ca}, y)$, where R is a simple role assertion, and p_1 and p_2 are two injective functions with disjoint ranges that map non-contextual roles to contextual roles. Alternatively, a new concept C_R is added for the role R, and the following assertions are added: $C_R(\mathbf{a}_\alpha^{Ca})$, $p_1(R)(\mathbf{a}_\alpha^{Ca}, x)$, and $p_2(R)(\mathbf{a}_\alpha^{Ca}, y)$.

Example 11. An *N-Ary relations* contextualization of our running example within the context CA of Ex. 2 contains the following axioms:
    $\mathtt{capitalOf\#1(babylon, rbcobe)}$
    $\mathtt{capitalOf\#2(rbcobe, babylonianEmpire)}$
    $\mathtt{validity(rbcobe, t)}$
    etc.

Singleton property is using a non-standard semantics of RDF but the same idea can be simulated with DL axioms. For each simple role axiom $R(x, y)$, the following axioms are added: $\mathbf{a}_\alpha^{Ch}(x, y)$ (that is, the term for the anchor is used as a role), $\{x\} \equiv \exists \mathbf{a}_\alpha^{Ch}.\{y\}$ (which guarantees that the anchor property is a singleton), and $\texttt{singletonPropertyOf}(\mathbf{a}_\alpha^{Ch}, R)$.

Example 12. A *Singleton Property* contextualization of our running example within the context *CA* of Ex. 2 contains the following axioms:

$\texttt{capital\#1}(\texttt{babylon}, \texttt{babylonianEmpire})$
$\texttt{singletonPropertyOf}(\texttt{capital\#1}, \texttt{capital})$
$\texttt{validity}(\texttt{capital\#1}, \texttt{t})$
etc.

The remaining approach, *NdFluents*, uses a similar approach as *NdTerms* except that it only renames the terms used as individuals and does not relativize the ontology. This ensures interesting properties wrt entailment preservation [15], but TBox axioms in different contexts are not distinguishable.

7 Discussion and Future Work

NdTerms and NdFluents are a concrete instantiations of a general approach of contextualizing (parts of) the terms in the ontology. Other instantiations would be possible, such as contextualizing role names (in a similar fashion as the singleton property), class names, or a combination of them. Then, *NdTerms* would be the approach where each and every term is contextualized, while in NdFluents only individuals are.

In the future, we would like to deepen the analysis of contextualization, filling gaps still present in this preliminary work. Especially, the combination of multiple annotations, or annotations of contextualized ontologies, present some interesting challenges. A more systematic comparison of the various approaches remains to be presented.

References

1. McCarthy, J.L.: Generality in Artificial Intelligence. Communications of the ACM **30**(12) (1987) 1029–1035
2. Giunchiglia, F.: Contextual Reasoning. Epistemologica **16** (1993) 345–364
3. Ghidini, C., Serafini, L.: Distributed First Order Logics. In Gabbay, D.M., de Rijke, M., eds.: Frontiers of Combining Systems 2. Volume 7 of Studies in Logic and Computation., Research Studies Press (2000) 121–139
4. Zimmermann, A.: Logical formalisms for Agreement Technologies. In Ossowski, S., ed.: Agreement Technologies. Springer (2013)

5. Kifer, M., Subrahmanian, V.S.: Theory of generalized annotated logic programming and its applications. Journal of Logic Programming **12**(3&4) (1992) 335–367
6. Udrea, O., Recupero, D.R., Subrahmanian, V.S.: Annotated RDF. ACM Transaction on Computational Logics **11**(2) (2010)
7. Zimmermann, A., Lopes, N., Polleres, A., Straccia, U.: A general framework for representing, reasoning and querying with annotated Semantic Web data. Journal of Web Semantics **11** (2012) 72–95
8. Carroll, J.J., Bizer, C., Hayes, P., Stickler, P.: Named graphs, provenance and trust. In Ellis, A., Hagino, T., eds.: Proceedings of the 14th International Conference on World Wide Web, WWW 2005, Chiba, Japan, May 10-14, 2005, ACM Press (May 2007) 613–622
9. Dividino, R.Q., Sizof, S., Staab, S., Schueler, B.: Querying for provenance, trust, uncertainty and other meta knowledge in RDF. Journal of Web Semantics **7**(3) (2009) 204–219
10. Hartig, O., Thompson, B.: Foundations of an Alternative Approach to Reification in RDF. CoRR **abs/1406.3399** (2014)
11. Hoffart, J., Suchanek, F.M., Berberich, K., Weikum, G.: YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia. Artificial Intelligence **194** (2013) 28–61
12. Brickley, D., Guha, R.V.: RDF Schema 1.1, W3C Recommendation 25 February 2014. W3C Recommendation, World Wide Web Consortium (February 25 2014)
13. Noy, N.F., Rector, A.L.: Defining N-ary Relations on the Semantic Web, W3C Working Group Note 12 April 2006. W3C Note, World Wide Web Consortium (April 12 2006)
14. Nguyen, V., Bodenreider, O., Sheth, A.: Don't like RDF reification?: making statements about statements using singleton property. In Chung, C.W., Broder, A.Z., Shim, K., Suel, T., eds.: 23rd International World Wide Web Conference, WWW '14, Seoul, Republic of Korea, April 7-11, 2014, ACM Press (April 2014) 759–770
15. Giménez-García, J.M., Zimmermann, A., Maret, P.: NdFluents: An Ontology for Annotated Statements with Inference Preservation. In Blomqvist, E., Maynard, D., Gangemi, A., Hoekstra, R., Hitzler, P., Hartig, O., eds.: The Semantic Web - 14th International Conference, ESWC 2017, Portorož, Slovenia, May 28 - June 1, 2017, Proceedings, Part I. Volume 10249 of Lecture Notes in Computer Science., Springer (May 2017) 638–654
16. Golbreich, C., Wallace, E.K.: OWL 2 Web Ontology Language, New Features and Rationale (Second Edition), W3C Recommendation 11 December 2012. W3C Recommendation, World Wide Web Consortium (December 11 2012)
17. Scott, D.S.: Identity and existence in intuitionalistic logic. In Fourman, M., Mulvay, C., Scott, D.S., eds.: Applications of Sheaves, Proceedings of the Research Symposium on Applications of Sheaf Theory to Logic, Algebra, and Analysis, Durham, July 9-21, 1977. Volume 753 of Lecture Notes in Mathematics., Springer (1979) 660–696
18. Cuenca-Grau, B., Kutz, O.: Modular Ontology Languages Revisisted. In Honavar, V.G., Finin, T.W., Caragea, D., Mladenic, D., Sure, Y., eds.: SWeCKa 2007: Proceedings of the IJCAI-2007 Workshop on Semantic Web for Collaborative Knowledge Acquisition, Hyderabad, India, January 7, 2007. (January 2007)
19. Giménez-García, A.Z.M.: Integrating Context of Statements within Description Logics. Technical report, Laboratoire Hubert Curien, Saint-Étienne, France (2017)
20. Goguen, J.A., Burstall, R.M.: Institutions: Abstract Model Theory for Specification and Programming. Journal of the ACM **39**(1) (1992) 95–146