The multirepresentation ontologies: a contextual description logics approach

Djamal Benslimane and Ahmed Arara

LIRIS, Université Lyon 1, IUT A Informatique
43, Bd. du 11 novembre 1918 69622 Villeurbanne Cedex, France
{djamal.benslimane,ahmed.arara}@iuta.univ-lyon1.fr

Abstract. Our motivation is to deal with the problem of multirepresentation in the context of ontology languages. In this short paper, we first propose a sub-language of DL as an ontology language and then we extend it by introducing some contextual constructors.

1 Introduction

Domain ontologies are constructed by capturing a set of concepts and their links according to a given context. A context can be viewed as various criteria such as the abstraction paradigm, the granularity scale, interest of user communities, and the perception of ontology developer. So, the same domain can have more than one ontology, where each one of them is described in a particular context. We call each one of these ontologies a MonoRepresentation ontology (MoRO) or a **non-contextual ontology**. Thus, concepts in MoRO are defined with one and only one representation. Our motivation is to see how we can describe ontology according to several contexts at the same time. We shall call such ontology a MultiRepresentation Ontology (MuRO) or a **contextual ontology**. A MuRO is an ontology that characterizes an ontological concept by a variable set of properties or attributes in various contexts. So, in MuRO, a concept is defined once with several representations such that a single representation is available for one context.

In this short paper, we will define a contextual ontology language to support multiple representations of ontologies. The underlying key idea of our work is to adapt the stamping mechanism proposed in [3] for the needs of multirepresentation in spatial databases to the needs of ontology languages. To achieve this goal, we first propose a sub-language of Description Logics DL [2] as an ontology language and then we extend it by introducing contextual constructors to allow multiple representations of concepts.

2 Syntactical and semantics aspects of (non-contextual) description logics

In this section, we give a brief description to the (non-contextual) description logics. DLs are a family of logics [1] designed to represent the taxonomic and conceptual knowledge of a particular application domain in an abstract and logical level. Starting from atomic concepts and roles, complex concepts (and roles) are built by using a set of constructors. The syntax and semantics of our non-contextual language are given below.

Definition 1. (Syntax of concept terms) *Let $\mathcal{C}$ be a set of concept names, $\mathcal{R}$ be a set of role names and $n \in \mathcal{N}$. Non-contextual concept terms C and D can be formed by means of the following syntax:*

$C, D \longrightarrow A \mid \top \mid \bot \mid$	(atomic, top and bottom concepts)
$C \sqcap D \mid C \sqcup D \mid \neg C \mid$	(conjunction, disjunction and complement)
$\exists R.C \mid \forall R.C \mid$	(existential quantification and value restriction)
$(\leq nR) \mid (\geq nR)$	(at most and at least number restriction)

Definition 2. (Semantics) *The semantics of the language is given by an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ which consists of an interpretation domain $\Delta^{\mathcal{I}}$, and an interpretation function $\cdot^{\mathcal{I}}$. The interpretation function $\cdot^{\mathcal{I}}$ maps each atomic concept $A \in \mathcal{C}$ to a subset $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and each role name $R \in \mathcal{R}$ to a subset $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. The extension of $\cdot^{\mathcal{I}}$ to arbitrary concepts is inductively defined as follows:*

$\top^{\mathcal{I}} = \Delta^{\mathcal{I}}; A^{\mathcal{I}} \subseteq \top^{\mathcal{I}}; (C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}; (C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$
$(\neg C)^{\mathcal{I}} = \top^{\mathcal{I}} \setminus C^{\mathcal{I}}; (\exists R.C)^{\mathcal{I}} = \{x \in \top^{\mathcal{I}} \mid \exists y : (x, y) \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$
$(\forall R.C)^{\mathcal{I}} = \{x \in \top^{\mathcal{I}} \mid \forall y : (x, y) \in R^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}}\}$
$(\leq nR)^{\mathcal{I}} = \{x \in \top^{\mathcal{I}} \mid \ \|\{y \mid (x, y) \in R^{\mathcal{I}}\}\| \leq n\}$
$(\geq nR)^{\mathcal{I}} = \{x \in \top^{\mathcal{I}} \mid \ \|\{y \mid (x, y) \in R^{\mathcal{I}}\}\| \geq n\}$

Let A be a concept name and let D be a concept term. Then $A = D$ is a terminological axiom (also called defined-as axiom). A terminology ($TBox$) is a finite set $\mathcal{T}$ of terminological axioms with the additional restriction that no concept name appears more than once in the left hand side of the definition. An interpretation $\mathcal{I}$ is a model for a $TBox$ $\mathcal{T}$ if and only if $\mathcal{I}$ satisfies all the assertions in $\mathcal{T}$. In this paper, we consider that ontologies are specialized by means of a set of terminological axioms. Hence, each ontology is a terminology.

3 Syntactical and semantics aspects of contextual description logics

For our requirements of multi-representation ontologies, we propose the notion of contextual concepts that may describe concepts associated with different contexts. Contextual concepts are derived basically from atomic concepts by using a set of non-contextual and/or contextual constructors. Contextual constructors are defined by specializing the non-contextual ones to allow the construction of a concept that is partially or completly available in some contexts. For the needs of multirepresentation, we add the following rule (expressed in definition 3) to the previously defined syntax given in definition 1 above.

Definition 3. (Syntax of contextual concept terms) *Let $s_1, \cdots, s_m$ be a set of context names. Contextual concept terms C and D can be formed by means of the following syntax:*

$C, D \longrightarrow$

$\exists_{s_1,\cdots,s_m} R.C \mid$	(contextual existential quantification)
$\forall_{s_1,\cdots,s_m} R.C \mid$	(contextual value restriction)
$(\leq_{s_1,\cdots,s_m} nR) \mid$	(contextual at most number restriction)
$(\geq_{s_1,\cdots,s_m} nR) \mid$	(contextual at least number restriction)
$C \sqcap_{s_1,\cdots,m} D$	(contextual conjunction)

The definition of non-contextual concepts remains always possible. Such concepts will exist in all contexts with a single representation. The semantics of the non-contextual language given in definition 2 is extended with the contextual notions. To define the semantics of the contextual constructors, we assume having a set of context names denoted as $S = \{s_1, s_2, ..., s_t\}$.

Definition 4. (Semantics of contextual concept terms) *The semantics of the contextual part of the language is given by a contextual interpretation defined in a context j over $\mathcal{S}$. A context j is either a simple context name belonging to $\mathcal{S}$ or a composed context defined as a conjunction of simple contexts*[1]*. A contextual interpretation $\mathcal{CI} = (\mathcal{I}_0, \mathcal{I}_1, \cdots, \mathcal{I}_j, \cdots, \mathcal{I}_t)$ is obtained by associating to each context j a non-contextual interpretation $\mathcal{I}_j = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}j})$, which consist of an interpretation domain $\Delta^{\mathcal{I}}$, and an interpretation function $\cdot^{\mathcal{I}j}$. The interpretation function $\cdot^{\mathcal{I}j}$ maps each atomic concept $A \in \mathcal{C}$ to a subset $A^{\mathcal{I}j} \subseteq \Delta^{\mathcal{I}}$ and each role name $R \in \mathcal{R}$ to a subset $R^{\mathcal{I}j} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. Let $Co(j)$ be a function which returns a set of all context names appearing in a simple/composed argument context j*[2]*. The extension of $\cdot^{\mathcal{I}j}$ to arbitrary concepts is inductively defined as follows:*

$(\exists_{s_1,\cdots,s_m} R.C)^{\mathcal{I}j} = \{x \in \Delta^{\mathcal{I}} \mid \exists y : (x, y) \in R^{\mathcal{I}j} \wedge y \in C^{\mathcal{I}j} \wedge Co(j) \cap \{s_1, \cdots, s_m\} \neq \emptyset\}$
$(\forall_{s_1,\cdots,s_m} R.C)^{\mathcal{I}j} = \{x \in \Delta^{\mathcal{I}} \mid \forall y : (x, y) \in R^{\mathcal{I}j} \rightarrow y \in C^{\mathcal{I}j} \wedge Co(j) \cap \{s_1, \cdots, s_m\} \neq \emptyset\}$
$(\leq_{s_1,\cdots,s_m} nR)^{\mathcal{I}j} = \{x \in \Delta^{\mathcal{I}} \mid Co(j) \cap \{s_1, \cdots, s_m\} \neq \emptyset \wedge \ \|\{y \mid (x, y) \in R^{\mathcal{I}j}\}\| \leq n\}$
$(\geq_{s_1,\cdots,s_m} nR)^{\mathcal{I}j} = \{x \in \Delta^{\mathcal{I}} \mid Co(j) \cap \{s_1, \cdots, s_m\} \neq \emptyset \wedge \ \|\{y \mid (x, y) \in R^{\mathcal{I}j}\}\| \geq n\}$
$(C \sqcap_{s_1,\cdots,s_m} D)^{\mathcal{I}j} = \{x \in \Delta^{\mathcal{I}} \mid x \in C^{\mathcal{I}j}\}$ *if* $Co(j) \cap \{s_1, \cdots, s_m\} = \emptyset$,
$= \{x \in \Delta^{\mathcal{I}} \mid x \in C^{\mathcal{I}j} \cap D^{\mathcal{I}j}\}$ *if* $Co(j) \cap \{s_1, \cdots, s_m\} \neq \emptyset$

A terminological axiom $A = D$ is satisfied by a contextual interpretation $\mathcal{CI}$ if $A^{\mathcal{I}j} = D^{\mathcal{I}j}$ for every context j. A contextual interpretation $\mathcal{CI}$ is a model for a $TBox$ $\mathcal{T}$ if and only if $\mathcal{CI}$ satisfies all the assertions in $\mathcal{T}$. Finally, a concept D subsumes a concept C iff $C \sqsubseteq D$ is satisfied in every contextual interpretation $\mathcal{CI}$. In fact, the contextual constructors can be defined as "necessarity" modal operator with limitation of the possible worlds to some given context names. Moreover, our work is based on the following asumptions: (a) constant domains, (b) finit domain, (c) rigid designator (or unique name asumption), and (d) the application of contextual constructors is limited to only concepts and not roles. Therefore, according to these asumptions and according to [4] with respect to the satisfiability analysis, the satisfiability of our proposed language is decidable.
The contextual value restriction constructor $(\forall_{s_1,\cdots,s_m} R.C)$ will define a new concept all of whose instances are related via the role R only to the individuals

[1] $j = s_1$ and $j = s_1 \wedge s_2$ are two examples of simple context and composed context respectively.

[2] $Co(s_1 \wedge s_2) = \{s_1, s_2\}$ and $Co(s_1) = \{s_1\}$ are two examples of the function Co.

of class C and in the contexts s_1 to s_m. For example, the expression *Employee* = $\forall_{s_1} EmployeeNumber.Number \sqcup \forall_{s_2} LastName.String$, in two contexts s_1 and s_2, defines the concept *Employee* as individuals with either an attribute $EmployeeNumber$ in context s_1 or an attribute $LastName$ in context s_2. Outside of the two contexts s_1 and s_2, the concept *Employee* corresponds to an empty set. The contextual existential quantification constructor $(\exists_{s_1,\cdots,s_m} R.C)$ will construct a new concept all of whose instances are related via the role R to at least one individual of type C and only in contexts s_1 to s_m. For example, the expression *Student*= $\exists_{s_1} Diploma.Graduate \sqcup \exists_{s_2} Play.Sport$ describes that student is an individual that has at least one graduation diploma in context s_1 or he participates in at least one sport in context s_2. It should be noted that the interpretation of the expression Student in the context s_1 or s_2 separately will give us an empty concept. The contextual conjunction $(C \sqcap_{s_1,\cdots,s_m} D)$ will define either a concept resulting of the conjunction of the two concepts C and D in the defined contexts s_1 to s_m, or a concept equivalent to concept C outside of all the given contexts (s_1 to s_m). For example, the expression $Modern - Manager = (Person \sqcap \forall Sex.Female) \sqcap_{s_1,s_2} (\exists_{s_1} Responsability.String \sqcup \exists_{s_2} Manage.Project)$ decsribes a modern Manager as being a person of sex female and who either has at least one responsability in the context s_1 or manages at least one project in the context s_2. Outside of the two contexts s_1 and s_2, Modern Manager is only defined as being a person of sex female and nothing else.

4 CONCLUSION

In this article, we have introduced an initiative formal approach for the multi-representation problem in ontology. We have extended a sub-language of DLs by introducing contextual constructors in order to define contextual concepts. Until now, we have not exploited the application of the proposed contextual constructors to terminological axioms, but il will be taken in consideration in futur. Finally, we intend to validate and test the proposed language in the domain of urbanism where we expect a wide range of contexts like transportation, land use and urban planing.

References

1. Franz Baader, Ian Horrocks, and Ulrike Sattler. Description logics as ontology languages for the semantic web. In *In Dieter Hutter and Werner Stephan, editors, Festschrift in honor of Jörg Siekmann, Lecture Notes in AI. Springer*, 2003.
2. M. Buchheit, F.M. Donini, and A. Scharaef. Decidable reasoning in terminological knowledge representation systems. *Journal of AI Research*, 1:109–138, 1993.
3. Stefano Spaccapietra, Christine Parent, and Christelle Vangenot. Gis databases: From multiscale to multirepresentation. In *4th Int. Symp., SARA 2000, Texas, USA. Proc. LNCS 1864 Springer.*, pages 57–70, July 2000.
4. F. Wolter and M. Zakharyaschev. Satisfiability problem in description logics with modal operators. In *In Proceedings of the 6th Int. Conf. on Knowledge Representation and Reasonning, KR'98, Trento, Italy, June 1998.*, pages 512–523, 2001.