Computing and Informatics, Vol. 32, 2013, 1001–1031, V 2019-Mar-4

NETWORKED ONTOLOGIES WITH CONTEXTUAL ALIGNMENTS

Abstract.

The problem of knowledge heterogeneity in the Semantic Web or in the context of information systems remains a major challenge for the scientific community, in particular when several ontologies developed independently and separately have to be exploited to exchange their knowledge. Several works have addressed the semantic heterogeneity issue in ontologies and proposed to align them with additional knowledge. Recently a formalism taking into account the challenge of applied techniques to represent and reason on aligned ontologies was proposed by the authors. The authors proposed a contribution that can be seen as an extension of existing work on the heterogeneous ontologies integration.

This formalism allows dealing with contextual representation and reasoning where ontologies and alignments by pairs of ontologies are developed in different and incompatible context. In this paper, some aspects of multi-level networked knowledge are recalled, detailing its semantics and discussing the comparison of the two semantics, DL-approach and DDL-approach, according to certain criteria, in order to measure their relevance and to give to readers a way to choose one semantics rather than another according to the context or the intended application.

Keywords: Networked knowledge semantics, contextual ontologies, contextual alignments.

1 INTRODUCTION

Recently, a Multi-Level Networked Knowledge (MLNK) formalism was proposed to allow contextualization of alignment representation [25] and [24]. This formalism attempts to solve the problem of alignments semantic heterogeneity using multiple alignment levels. This favours dealing with the alignment complexity going up in abstraction instead of trying to force alignment experts to provide coherent alignments at the lowest level of detail (increasingly hard as networks grow due to the cognitive limits of humans). Syntactically, this formalism is defined in a very general way and is independent of the ontologies underlying logic, exploiting the recursive technique to build a hierarchically structured knowledge base in levels. An instantiation of the generic formalism was evoked, with the interest put on OWL ontologies.

In the literature, one may find three basic semantic languages for the interpretation of Network of Aligned Ontologies: Non-Contextual And Centralized Semantics; Contextual And Distributed Semantics; and Contextual And Integrated Semantics. But none of those semantics can be applied directly for interpreting the MLNK formalism.

Inspired by those, this paper proposes an extended semantics for the interpretation of Network of Aligned Ontologies on several levels. The advantage of the extended semantics lies in the fact that each alignment expressed between a source and target ontology is independently treated, as each one possesses its own distinct vocabulary and semantics. The first proposed semantic, Extended Non-Contextual And Centralized Semantics (ENCACS), favours the fact that ontologies and the alignment set expressed in pairs are heterogeneous, either expressed in the same context or different compatible ones. The second proposed semantic, Distributed And Contextual-On-Several-Levels Semantics (DACOSLS), is defined in order to support ontologies and alignments heterogeneity, even if those are expressed in distinct and incompatible contexts. This semantic favours the contextualization of ontologies as well as alignments.

The DL-approach applies Extended Non-Contextual And Centralized Semantics (ENCACS), which was developed and implemented with the obtained results presented in [25].

The approach applying Distributed And Contextual-On-Several-Levels Semantics (DACOSLS) was developed and presented in a previous work [24]. In the present one, the approach concepts are recalled, then we describe the prototype used to reason on the MLNK following the DDL-approach. Results from our test protocol are presented and compared with the DL-approach prototype results.

In order to show the difference between the different approaches, a thorough comparison is made in this paper. The resulting comparative study focuses on the adaptability of one semantic over another. This will allow readers to justify the choice of either for a given application, taken into account its context.

The organization of the rest of the article is as follows: In Section 2 the notion of networked ontologies while highlighting the specific definitions to multi-level networked knowledge is recalled. Section 3 describes the semantic approaches of MLNK interpretation. Section 4 provides detailed information on the implementation of the DDL-based MLNK reasoner prototype. In Section 5, the semantic approaches (DL-approach and DDL-approach) are compared showing how they are different from each other and in which cases one is more interesting than the other. Section 6 gives a synthesis of related works and discussion. Finally, Section 7 addresses a general conclusion.

2 NETWORK OF ALIGNED ONTOLOGIES

Generally, Network of Aligned Ontologies (NAO) formalisms were introduced with one or more motivations. Syntactically, they are composed of a family of local ontologies and alignments that bind them. They are endowed with one or more semantics for possible reasoning on aligned knowledge.

In this section, formalisms that can handle reasoning on NAOs are presented with their motivations, syntactic and semantic representations. Table 6 summarizes the latter, presenting motivations, syntax and semantics of the formalisms described in this paper.

2.1 Motivation

We start by identifying the different motivations behind existing formalisms, then we define the motivation for the introduction of Multi-Level Networked Knowledge.

2.1.1 Motivations behind Network of Aligned Ontologies

There are four important motivations associated with NAOs:

Ontology combination: this motivation is favoured to combine several non-heterogeneous ontologies, where each one describes a separated, very different viewpoints and complementary portions of a complex domain. In general, *links* are used in order to link entities belonging to different ontologies (*e.g.*, $\mathtt{O}_1$:`France` $\overset{\textbf{is-part-of}}{\rightarrow}$ $\mathtt{O}_2$:`Europe`) (see Section 2.2.2). As an example, $\mathcal{E}$-connection [27] is a formalism proposing a syntactic representation and a formal semantics for reasoning on a Network of Aligned Ontologies, where entities in different ontologies are connected by *links*.

Resolution of semantic heterogeneity between ontologies: in order to resolve the semantic heterogeneity problem between ontologies. It is necessary to use ontology *mappings* which are semantic relations between entities (*e.g.*, $\mathtt{O}_1$:`java` $\overset{\perp}{\rightarrow}$ $\mathtt{O}_2$:`java`). As an example, DDL formalism [8] proposes a syntactic and semantic representation that permits reasoning on a Network of Aligned Ontologies using *mappings*.

Ontology import: The import of ontologies is mainly used to promote the reuse of the concepts, roles or individuals defined in other ontologies. The notion of importing entities belonging to other ontologies with the goal of reusing them was introduced in [7]. This is mainly interesting, as it permits reusing a number of entities from a given ontology without importing it as a whole.

Mediation of alignment: This motivation ensures an independent management of the alignments. As an example, one may cite the alignments composition for exchanging and a better reusing of the latter through the network of ontologies. The main goal is still to reuse existing alignments in order to obtain newer ones. The IDDL formalism [33] proposes a syntactic and semantic representation in order to manage and exploit alignments to ensure mediation through the knowledge network.

2.1.2 Motivation behind MLNK

The set of pair ontology alignments have their own vocabulary. They are developed independently from each other by domain experts with different viewpoints, being then possibly heterogeneous. In order to solve the heterogeneity problem between alignments, the latters, need to be linked in the higher levels.

A real-life application example of gas turbine ontological representation is presented. Due to their wide usage in electricity production, the gas turbine is often found in the center of large power systems that need to be managed in terms of knowledge and maintenance. Four ontologies describing gas turbine have been developed for the purpose of this example, namely:

- an ontology for equipment (`eq`), modelling the turbine technical and hierarchical knowledge. This information is provided by the constructor and contains 5033 concepts, where each concept describes an equipment or turbine component, such as the concept `flame-detector` given by instance $\mathtt{FD}_1$;
- an ontology termed (`Pr`), modelling spare parts, such as the concept `trim` given by the instance $\mathtt{T}_1$;
- an ontology for modelling the position of the equipment in the turbine hierarchy (`zn`);
- An ontology created from an existing database `mt`, using a semi-automatic approach, covering, maintenance operations (both preventive and currative). The `mt` ontology exploits the first ontologies (`eq`),`Pr` and `zn`) in order to provide details on equipments and spare parts concerned by maintenance operations.

These ontologies are independent and heterogeneous; we aim to exploit them via a common interface without constraining or altering their internal representation. We propose for that effect, to insert ontology alignments separately without favouring any of the local representations. Correspondences of the *mappings* type are produced via independent tools, the case for the following correspondences: $\mathtt{mt{:}belong} \xleftrightarrow{\perp} \mathtt{eq{:}belong}$ **between** $(\mathtt{mt}, \mathtt{eq})$ **ontologies pair and** $\mathtt{pr{:}trim} \xleftrightarrow{\sqsubseteq} \mathtt{eq{:}instrumentation}$ **between** $(\mathtt{pr}, \mathtt{eq})$**. The set of produced *mappings* may be enriched semi-automatically by new links (terms linking two different ontologies). This operation is performed by experts, understanding one expert for each ontologies pair. Alignments are then developed independently by domain expert expressing different viewpoints. It is then observed that the semantic heterogeneity problem occurs at the alignment level. It is**

the case for alignments $A_{\mathtt{pr\text{-}eq}}$ and $A_{\mathtt{eq\text{-}zn}}$, with the terms $A_{\mathtt{pr\text{-}eq}}$:compose and $A_{\mathtt{eq\text{-}zn}}$:part-of, these *links* have similar semantics. In order to infer knowledge, it is necessary to insert an equivalence relation between the two *links* $A_{\mathtt{pr\text{-}eq}}$:compose and $A_{\mathtt{eq\text{-}zn}}$:part-of. This comes to align ontology alignments.

Example 1. An excerpt of ontologies and associated alignments are presented in Table 1.

Ontologies	Axioms
`eq`:	$\mathtt{flame\text{-}detector}(\mathtt{FD}_1)$ $\mathtt{flame\text{-}detector} \sqsubseteq \exists\mathtt{belong.instrumentation}$
`pr`:	$\mathtt{trim}(\mathtt{T}_1)$
`zn`:	$\mathtt{zone}(\mathtt{ANNA1TG01})$
`mt`:	$\mathtt{intervention}(\mathtt{I}_1)$ $\mathtt{team}(\mathtt{TE}_1)$ $\mathtt{intervene}(\mathtt{TE}_1, \mathtt{I}_1)$ $\mathtt{member} \sqsubseteq \exists\mathtt{belong.team}$
Alignments	
$A_{\mathtt{eq\text{-}zn}}$:	$\mathtt{eq{:}FD}_1 \overset{\textbf{part-of}}{\longleftrightarrow} \mathtt{zn{:}ANNA1TG01}$
$A_{\mathtt{pr\text{-}eq}}$:	$\mathtt{pr{:}trim} \overset{\sqsubseteq}{\longleftrightarrow} \mathtt{eq{:}instrumentation}$ $\mathtt{pr{:}T}_1 \overset{\textbf{compose}}{\longleftrightarrow} \mathtt{eq{:}FD}_1$
$A_{\mathtt{mt\text{-}eq}}$:	$\mathtt{mt{:}I}_1 \overset{\textbf{concern}}{\longleftrightarrow} \mathtt{eq{:}FD}_1$ $\mathtt{eq{:}belong} \overset{\perp}{\longleftrightarrow} \mathtt{mt{:}belong}$
$A_{\mathtt{A_{pr\text{-}eq}\text{-}A_{eq\text{-}zn}}}$:	pr-eq:**compose** $\overset{\equiv}{\longleftrightarrow}$ eq-zn:**part-of**

Table 1. An excerpt of ontologies and associated alignments

In order to solve the heterogeneity problem occurring between alignments vocabularies, alignment at a higher level is proposed. This, however, necessitates the introduction of a formalism permitting a representation of MLKN. Figure 1 represents the turbine example showing alignment levels.

None of the existing formalisms treats alignments separately and independently with respect to ontologies and the other alignments. As a result, no proposition was made to align alignments, making all existing formalisms not able to support alignments contextualization.

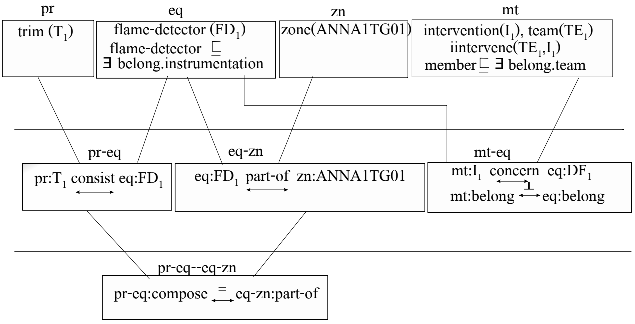

Fig. 1. Knowledge representation levels

2.2 The Network of Aligned Ontologies syntax

A network of aligned ontologies is composed of a family of local ontologies also called modular ontologies or source knowledge bases and a family of alignments. Knowledge node is a new concept defined to formalize MLNK syntax.

2.2.1 Local ontology

The local ontologies $\{O_i\}$ of a network of aligned ontologies are indexed by a finite set of indices I. Ontologies are developed and designed in different contexts. The notion of information context has been extensively discussed in several works like [28], [16] and recently [35], with a general definition of the context being a given "point of view" or "provenance" or even "a temporal valid information". Each ontology O_i is represented in a knowledge representation language defined by:

- A syntax, that is a set of symbols and sentences (or formulas) that can be built with them;
- A notion of interpretations, which defines a domain of interpretation and associate symbols with structures over the domain;
- A satisfaction relation, which relates interpretations to the sentences they satisfy.

There are many languages for knowledge representation applied to local ontologies definition, one may cite First-Order Logic, Modal Logic, Description Logic, etc.

The proposed syntax for MLNK is generic and independent from any ontologies language (See Section 2.2.3). In order to interpret it, the choice of existing logic is given to the user, such as First-Order Logic, Modal logic, DL, etc. In the presented work we focused on DL ontologies, as DL is fundamental for semantic web and OWL ontologies. Table 6, resumes local ontologies languages for existing formalisms.

Let us recall some basics formulation and concepts of DL [5] that will be used for the remainder of the paper.

DL ontology is composed of concepts, roles and individuals, as well as axioms built out of these elements. A concept is either a primitive concept A, or, given concepts C, D, role R, individuals $a_1, \dots, a_k$, 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 $\{a_1, \dots, a_k\}$. A role is either a primitive role P, or, given roles R and S, $R \sqcup S$, $R \sqcap S$, $\neg R$, R^-, $R \circ S$ and R^+.

Interpretations are pairs $\langle \Delta^I, \cdot^I \rangle$, where Δ^I is a non-empty set (the domain of interpretation) and $\cdot^I$ is the function of interpretation such that for all primitive concepts A, $A^I \subseteq \Delta^I$, for all primitive roles P, $P^I \subseteq \Delta^I \times \Delta^I$, and for all individuals a, $a^I \in \Delta^I$.

Interpretations of complex concepts and roles is inductively defined by $\bot^I = \emptyset$, $\top^I = \Delta^I$, $(C \sqcup D)^I = C^I \cup D^I$, $(C \sqcap D)^I = C^I \cap D^I$, $(\exists R.C)^I = \{x | \exists y.y \in C^I \wedge \langle x, y \rangle \in R^I\}$, $(\forall R.C)^I = \{x | \forall y. \langle x, y \rangle \in R^I \Rightarrow y \in C^I\}$, $(\leq nR.C)^I = \{x | \sharp\{y \in C^I | \langle x, y \rangle \in R^I\} \leq n\}$, $(\geq nR.C)^I = \{x | \sharp\{y \in C^I | \langle x, y \rangle \in R^I\} \geq n\}$, $(\neg C)^I = \Delta^I \setminus C^I$, $\{a_1, \dots, a_k\} = \{a_1^I, \dots, a_k^I\}$, $(R \sqcup S)^I = R^I \cup S^I$, $(R \sqcap S)^I = R^I \cap S^I$, $(\neg R)^I = (\Delta^I \times \Delta^I) \setminus R^I$, $(R^-)^I = \{\langle x, y \rangle | \langle y, x \rangle \in R^I\}$, $(R \circ S)^I = \{\langle x, y \rangle | \exists z. \langle x, z \rangle \in R^I \wedge \langle z, y \rangle \in S^I\}$ and $(R^+)^I$ is the reflexive-transitive closure of R^I.

Axioms are either subsumption $C \sqsubseteq D$, sub-role axioms $R \sqsubseteq S$, instance assertions $C(a)$, role assertions $R(a, b)$ and individual identities $a = b$, where C and D are concepts, R and S are roles, and a and b are individuals. An interpretation I *satisfies* axiom $C \sqsubseteq D$ if and only if $C^I \subseteq D^I$; it satisfies $R \sqsubseteq S$ if and only if $R^I \subseteq S^I$; it satisfies $C(a)$ if and only if $a^I \in C^I$; it satisfies $R(a, b)$ if and only if $\langle a^I, b^I \rangle \in R^I$; and it satisfies $a = b$ if and only if $a^I = b^I$. When I satisfies an axiom α, it is denoted by $I \models \alpha$.

An ontology O is composed of a set of terms (primitive concepts/roles and individuals) called the signature of O and denoted by $\mathrm{Sig}(O)$, and a set of axioms denoted by $\mathrm{Ax}(O)$. An interpretation I is a model of an ontology O if and only if for all $\alpha \in \mathrm{Ax}(O)$, $I \models \alpha$. In this case, we write $I \models O$. 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 $I \in \mathrm{Mod}(O)$, $I \models \alpha$.
An ontology is logically consistent if the ontology has a model.

2.2.2 Alignments

The correspondences represent relations between entities (terms or formulas) belonging to different ontologies. The set of correspondences is termed ontology alignment. Let us recall that there are two types of correspondences:

- The first type of alignment (*mapping*) concerns the correspondences which are associated with a predefined set of relations such as subsumption, equivalence, disjunction, etc. where the given semantic is fixed for all interpretations (*e.g.*, $\mathtt{O_1{:}java} \xleftrightarrow{\perp} \mathtt{O_2{:}java}$). Which means that the `java` entity in the ontology O_1 is semantically different from the `java` entity in O_2.
- The second type of alignment (*links*) is used to link ontologies covering complementary domains, it is the case of $\mathcal{E}$-connection [27], $E - \mathcal{SHIQ}$ [32] and

MLNK [25]. It is represented by inter-ontological roles between entities, termed simply links (*e.g.*, $\mathtt{O_1{:}France} \xleftrightarrow{\textbf{is-part-of}} \mathtt{O_2{:}Europe}$).

The syntax representation of correspondences, differs from one formalism to another. As an example, DDL [8] is cited here, where *mappings* (DDL does not handle *links*) are represented by directional arrows expressed as the target ontology point of view (*e.g.*, $\mathtt{O_1{:}A} \xrightarrow{\sqsubseteq} \mathtt{O_2{:}B}$) Where the inverse of the correspondence (*e.g.*, $\mathtt{O_2{:}B} \xrightarrow{\sqsupseteq} \mathtt{O_1{:}A}$) is not valid. In the case of the proposed formalism, as well as for IDDL, double arrows are used to express correspondences with an external "point of view" of target and source ontologies, (*e.g.*, $\mathtt{O_1{:}A} \xleftrightarrow{\sqsubseteq} \mathtt{O_2{:}B}$), Where the inverse correspondence (*e.g.*, $\mathtt{O_2{:}B} \xleftrightarrow{\sqsupseteq} \mathtt{O_1{:}A}$) is valid and can be inserted. However, IDDL express correspondences from a global point of view with respect to the whole ontology network. This is quite difficult to achieve, considering the limited expert's knowledge not allowing a complete understanding of all domain aspects. MLNK suggest expressing correspondences according to a global point of view with respect to a pair of ontologies.

Definition 2.1 (Initial alignment language representation)**.** The alignment language L_A that allows expressing correspondences is initially defined as a pair $\langle E, R\rangle$ where E is a function from any ontology $O \subseteq L_A$ which defines the matchable entities of ontology O and R a set of symbols that allow relating these entities, with $R = \{\sqsubseteq, \equiv, \perp, \in, =\}$ [14].

Alignment language, in this case, is reduced to the terms of existing vocabularies and does not have its own vocabulary.

Definition 2.2. A correspondence expressed in this language L_A is given by a triplet $\langle e_1, r, e_2\rangle$ noted $e_1 \xrightarrow{r} e_2$ where e_1, e_2 are entities belonging respectively to $E(O_1), E(O_2)$ and $r \in R$ or r is a *link*.

These definitions do not constitute a problem if all correspondences are of *mapping* types, on the other hand, if some of them are *mappings* and others are *links*, the problem arises necessarily. This is due to the fact that the links are terms likely to have several interpretations, and can vary from one pair of ontologies to another.

The previous definitions of alignment language and correspondences do not permit alignment contextualization. To remedy to the problem, recent definitions have been given where the alignment language has its own vocabulary allowing to express distinctly *mappings* and *links*.

Definition 2.3. Proposed alignment language: An alignment language L_A permits the description of correspondences between two vocabularies. It is also characterized by a syntax (how correspondences are expressed) and a semantic (how correspondences are interpreted). The syntax of L_A is defined by:

- a set of terms, called links, specific to the alignment language noted $V(L_A)$;

- a function $E(L_A)$, which associate to each signature of a representation language L, a set of entities that can be aligned;
- a set of relation's symbols $R(L_A)$.

Thus, the syntax of an alignment language L_A is defined by the triple $\langle V(L_A), E(L_A), R(L_A)\rangle$, denoted $\langle V, E, R\rangle$ when no ambiguity exists. Two types of correspondences might be defined as *mapping* and *link* correspondences.

Definition 2.4 (*mapping* correspondence)**.** Let V_1 and V_2 be two aligned vocabularies and let the triplet $\langle V, E, R\rangle$ denotes an alignment language. A *mapping* correspondence is a triple $\langle e_1, e_2, r\rangle$ noted $e_1 \overset{r}{\longleftrightarrow} e_2$ where:

- $e_1 \in E(V_1)$ and $e_2 \in E(V_2)$ are matchable entities;
- $r \in R$ denotes a relation that holds between e_1 and e_2 with $R = \{\sqsubseteq, \equiv, \perp, \in, =\}$.

Definition 2.5 (*link* correspondence)**.** Let us consider V_1 and V_2 two aligned vocabularies and $\langle V, E, R\rangle$ an alignment language. A *link* correspondence is a formula in the form $e_1 \overset{l}{\longleftrightarrow} e_2$ where:

- $e_1 \in E(V_1)$ and $e_2 \in E(V_2)$ are matchable entities;
- $l \in V$ denotes a relation that holds between e_1 and e_2.

Definition 2.6 (Alignment)**.** Let V_1 and V_2 be two vocabularies. An *alignment of* V_1 *and* V_2 is a tuple $\Lambda = \langle V, \kappa, \lambda\rangle$ where:

- V is an alignment vocabulary;
- κ is a set of *mapping* correspondences, $e_1 \overset{r}{\longleftrightarrow} e_2$ where $e_1 \in E(V_1)$, $e_2 \in E(V_2)$ and $r \in R$;
- λ is a set of *link* correspondences, $e_1 \overset{l}{\longleftrightarrow} e_2$ where $e_1 \in E(V_1)$, $e_2 \in E(V_2)$ and $l \in V$;

2.2.3 Knowledge node

The syntactic formalization of MLNK is defined in a very general way, independently of any language, using a recursion technique to build a knowledge base, hierarchically structured in levels. In other words, it is composed of a family of knowledge nodes and alignments between any pair of nodes where each node is it self-composed of a pair of aligned sub-nodes. Hence a dynamic construction of knowledge nodes where the most elementary node is an ontology. An ontology is therefore, a level 0 knowledge node, while all knowledge node of level $m > 0$, is constructed from a number of nodes from an inferior level, linked using alignment. Formally the node is defined as:

Definition 2.7 (Knowledge node)**.** A *knowledge node* is a pair $K = \langle V_K, A_K\rangle$ where V_K is a vocabulary, also written Voc(K) and both V_K and A_K are defined recursively:

- an ontology O is a knowledge node with vocabulary $\mathrm{Voc}(O) = \mathrm{Sig}(O)$ and A_K is the set of axioms;
- for $n \geq 1$, if $K_1, \ldots, K_n$ are knowledge nodes with vocabularies $\mathrm{Voc}(K_1), \ldots, \mathrm{Voc}(K_n)$, and for all $i, j \in [1, n]$, Λ_{ij} is an alignment of $\mathrm{Voc}(K_i)$ and $\mathrm{Voc}(K_j)$, then $K = \langle V_K, A_K \rangle$ is a knowledge node with the vocabulary:

$$V_K = \bigcup_{i,j \in [1,n]} \{ij : l \mid l \in \mathrm{Voc}(\Lambda_{ij})\} \cup \bigcup_{i \in [1,n]} \{i : e \mid e \in \mathrm{Voc}(K_i)\}$$

  and $A_K = \langle (K_i)_{i \in [1,n]}, (\Lambda_{ij})_{i,j \in [1,n]} \rangle$.

If a knowledge node includes only ontologies and ontology alignments, we call it a Network of Aligned Ontologies. If a knowledge node is neither a single ontology nor a network of aligned ontologies, we call it a Multi-Level Networked Knowledge base.

2.3 The Network of Aligned Ontologies semantics

Three basic semantics associated are defined in [34] to Network of Aligned Ontologies. Two other extended semantics inspired by basic semantics are presented in what follows.

2.3.1 Non-Contextual And Centralized Semantics (NCACS)

This semantic is formalized by classical logic, there is a unique interpretation domain for the whole network which is the union of all local interpretation domains (Δ_i for all $i \in [1, n]$). Interpretation is a model if it satisfies all the axioms of local ontologies (O_i for all $i \in [1, n]$) and alignments (A_{ij} for all $i, j \in [1, n]$). See Figure 2.

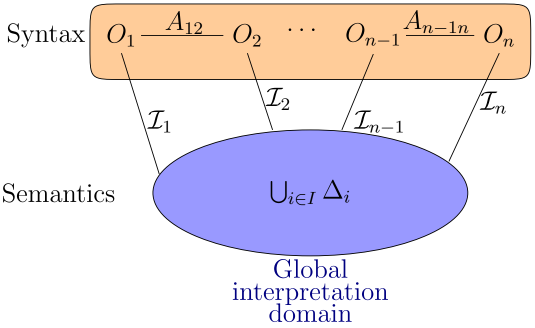

Fig. 2. Non-Contextual And Centralized Semantics (NCACS)

2.3.2 Contextual And Distributed Semantics (CADS)

There are two variants of CADS:

- Variant 1: this semantic is formalized by distinct and separate local interpretations ($\mathcal{I}_i$ for all $i \in [1, ..n]$), but linked by domain relations (r_{ij} for all $i, j \in [1, ..n]$). The distributed interpretation $\mathcal{I}$ is composed of local interpretations and domain relationships, $\mathcal{I} = \langle \{\mathcal{I}_i\}, \{r_{ij}\} \rangle$ for all $i, j \in [1, ..n]$. It is a model of the network if (see Figure 3):
  - Each local interpretation $\mathcal{I}_i$ satisfies the axioms of the corresponding ontology (O_i for all $i \in [1, ..n]$);
  - The local interpretations and the domain relationships satisfy the constraints imposed by the alignments (A_{ij} for all $i, j \in [1, ..n]$).
- Variant 2: this semantic is formalized by distinct and separate local interpretations ($\mathcal{I}_i$ for all $i \in [1, ..n]$), and a special interpretation ($\mathcal{I}_{ij}$ for all $i, j \in [1, ..n]$) assigns to each *link* R^{ij} from i to j a domain relation, that is, a subset of $\Delta_i \times \Delta_j, i, j \in [1, ..n]$. The combined interpretation $\mathcal{I}$ is composed of local interpretations and a special interpretations, $\mathcal{I} = \langle \{\mathcal{I}_i\}, \{\mathcal{I}_{ij}\} \rangle$ for all $i, j \in [1, ..n]$. It is a model of the network if:
  - Each local interpretation $\mathcal{I}_i$ satisfies the axioms of the corresponding ontology (O_i for all $i \in [1, ..n]$);
  - The local interpretations and the special interpretation satisfy the constraints imposed by the alignments (A_{ij} for all $i, j \in [1, ..n]$).

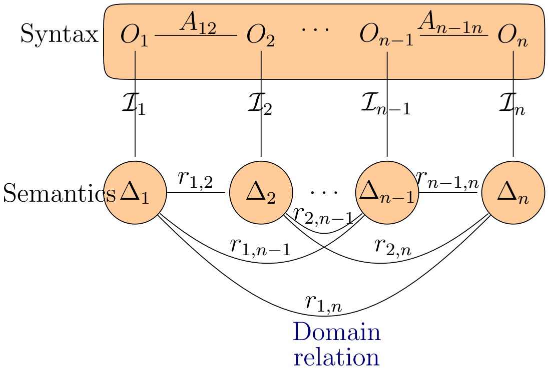

Fig. 3. Contextual And Distributed Semantics (CADS)

2.3.3 Contextual And Integrated Semantics (CAIS)

CAIS can be seen as, the combination of centralized semantics (on the alignment side) and distributed semantics (on the local ontologies side). The local interpretations are distinct and separate but not directly related. They are connected by means of the equalizing functions to an additional interpretation domain. The equalizing function is a projection function from local interpretation domain to a virtual global domain. The global domain is used to interpret inter-ontological knowledge (alignment) from a global point of view. It is the first idea that defines an independent interpretation of the alignments but the centralization of the alignment interpretation in a single additional domain does not allow alignment contextualization. Distributed, integrated interpretation is composed of local interpretations and equalizing functions $\mathcal{I} = \langle \{\mathcal{I}_i\}, \epsilon_i \rangle$ for all $i \in [1, ..n]$ (see Figure 4). It is a model of the network if:

- Local interpretations $\mathcal{I}_i$ satisfy source ontologies (O_i for all $i \in [1, ..n]$);
- The pairs of source interpretation and target with the equalizing functions (ϵ_i for all $i \in [1, ..n]$) satisfy the constraints imposed by the alignments (A_{ij} for all $i, j \in [1, ..n]$).

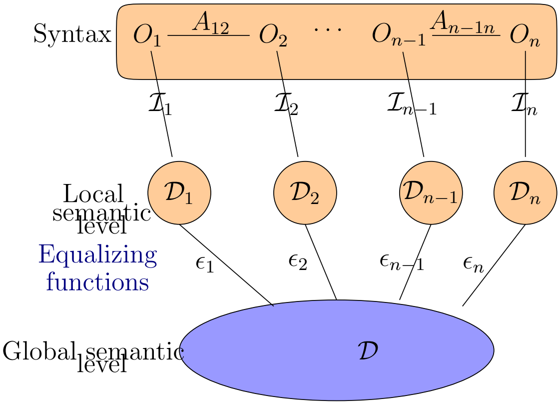

Fig. 4. Contextual And Integrated Semantics (CAIS)

Inspired from the basic semantics, an extended semantics for the interpretation of Network of Aligned Ontologies on several levels is proposed. The proposed semantics, have the ability to support independent alignment interpretations as well as their contextualization.

2.3.4 Extended Non-Contextual And Centralized Semantics (ENCACS)

Extended Non-Contextual And Centralized Semantics considers that the set of ontologies with corresponding alignments are interpreted in a single domain. The interpretation domain is the result of the union of the existing interpretation domains consisting of ontologies and alignments. An interpretation is a model of the network if it satisfies all the axioms of local ontologies and alignments. These solutions are adapted for the integration of independent ontologies, independently aligned and developed in different but compatible and not contradictory contexts. Figure 5 shows an extension of the centralized semantics with the integration of alignment interpretation.

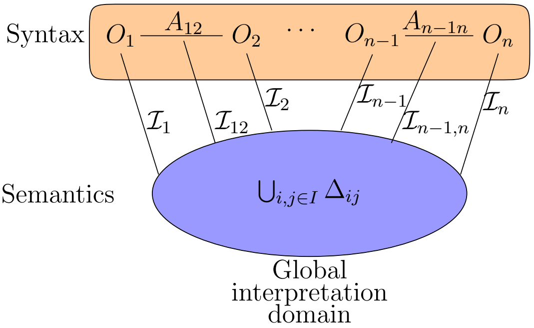

Fig. 5. Extended Non-Contextual And Centralized Semantics (ENCACS)

2.3.5 Distributed And Contextual-On-Several-Levels Semantics (DACOSLS)

This semantic is an extension of CADS semantics, where alignments are interpreted from the target ontology point of view. In order to interpret alignments of the source and target ontologies independently, the idea is to generate an alignment-interpretation domain (see Figure 6). Then, local interpretations are related to alignment-interpretations through domain relationships. The notion of independent alignment-interpretations by a pair of ontologies which ensures the contextualization of the alignments. A distributed interpretation is a model if:

- The local interpretations satisfy the local ontologies;
- The alignment-interpretations satisfies the constraints posed by the alignments;
- The local interpretation, the alignment-interpretations with the domain relations satisfies the contradictions posed by the equivalence bridge rules.

3 SEMANTIC APPROACHES

Two semantic approaches are usually associated with MLNK. DL-approach is defined to interpret and reason on multi-levels networked ontologies according to EN-

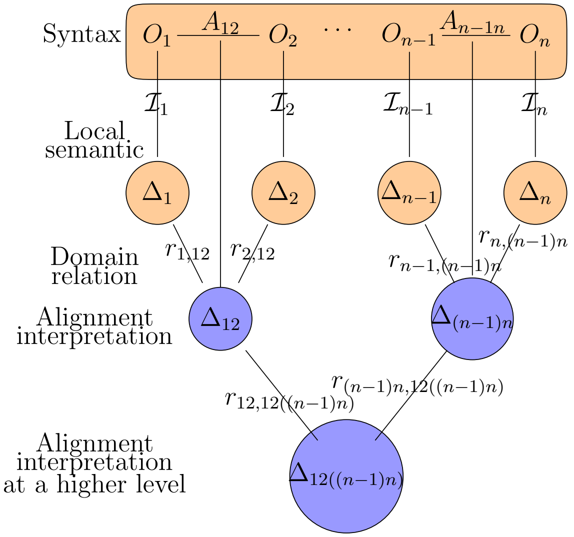

Fig. 6. Distributed And Contextual-On-Several-Levels Semantics (DACOSLS)

CACS (see [25] for more details). Where the DDL-approach is defined to interpret and reason on multi-levels networked ontologies according to Distributed and Contextual-on-several-levels Semantics.

3.1 DL-approach

This approach consists in the transformation of the multi-levels networked ontologies into a unique description logic ontology "DL-ontology" following the steps below:

- Prefix the ontologies which consist in assigning the indexes of the source ontologies to their corresponding entities;
- Transformation of alignment into description logic axioms "DL-axioms";
- Generation of the global ontology, also known as a multi-level knowledge node, obtained recursively by the union of the source ontologies with the integration of the axioms originating from alignments;
- testing the MLNK consistency through the DLMLNKR prototype.

3.2 DDL-approach: syntax and semantics

This approach consists in the transformation "SystDis" of the multi-levels networked ontologies to a DDL system, following the steps below:

1. Generation of alignment-ontology;
2. Generation of equivalence bridge rules between terms of alignment-ontology and terms belonging to corresponding source ontologies.

Let us recall the necessary definitions, in order for the reader to better understand implementation details of the DDLMLNKR prototype presented in section 4.

Definition 3.1 (Indexing the ontology element)**.** Let i be an index. We define the function prefix on the terms, axioms and ontologies, such that $\mathsf{prefix}(X, i) = \{i{:}X\}$ when X is an atomic concept, atomic role or an individual, and if X is a formula, $\mathsf{prefix}(X, i)$ is a formula where all terms are prefixed by i.

Definition 3.2 (Alignment-ontology signature)**.** Let us consider a multi-level knowledge node K, alignment-ontology signature Σ_A is defined as follows according to the case:

- if K is an ontology then $\Sigma_A = \emptyset$;
- if K a multi-level knowledge node composed of sub nodes $K_1, \ldots, K_n$ and A_{ij} which is alignment between K_i and K_j for $i, j \in [1, n]$, then:

$$\Sigma_A(K) = \bigcup_{i,j\in[1,n]} \{\mathsf{prefix}(X, i), \mathsf{prefix}(Y, j) \mid i{:}X \xleftrightarrow{r} j{:}Y \in A_{ij}\} \cup \bigcup_{i,j\in[1,n]} \mathrm{Voc}(A_{ij})$$

where X and Y are the concepts, roles or individuals and $r \in \{\sqsubseteq, \equiv, \perp, \in, =\}$, and $\mathrm{Voc}(A_{ij})$ means the alignment vocabulary, the *links* of A_{ij}.

Alignment-ontology formulas are the set of generated formulas from correspondences. Firstly, the function associating each correspondence to an axiom is defined.

Definition 3.3 (Correspondence transformation into axioms)**.** Let us consider an alignment A_{ij} between a node i and a node j, for $i, j \in [1, n]$. We define trans a function which assigns to each correspondence of A_{ij} a DL axiom: $\mathsf{trans}(\{i{:}A \xleftrightarrow{\sqsubseteq} j{:}B\}) = \{\mathsf{prefix}(A, i) \sqsubseteq \mathsf{prefix}(B, j)\}$; $\mathsf{trans}(\{i{:}A \xleftrightarrow{\equiv} j{:}B\}) = \{\mathsf{prefix}(A, i) \equiv \mathsf{prefix}(B, j)\}$; $\mathsf{trans}(\{i{:}A \xleftrightarrow{\perp} j{:}B\}) = \{\mathsf{prefix}(A, i) \sqsubseteq \neg\mathsf{prefix}(B, j)\}$; $\mathsf{trans}(\{i{:}u \xleftrightarrow{\in} j{:}A\}) = \{\mathsf{prefix}(A, j)(i{:}u)\}$; $\mathsf{trans}(\{i{:}u \xleftrightarrow{=} j{:}u'\}) = \{i{:}u = j{:}u'\}$; $\mathsf{trans}(\{i{:}u \xleftrightarrow{l} j{:}u'\}) = \{\mathsf{role}(l)(i{:}u, j{:}u')\}$; $\mathsf{trans}(\{i{:}A \xleftrightarrow{l} j{:}B\}) = \{\mathsf{prefix}(A, i) \sqsubseteq \exists\mathsf{role}(l).\mathsf{prefix}(B, j)\}$, where A, B, u and u' are the matchable entities and l is a *link*.

Definition 3.4 (Alignment-ontology formulas)**.** Let us consider a multi-level knowledge node K, the set of alignment-ontology formulas F_A is defined, according to the cases as follows:

- if K is an ontology then $F_A = \emptyset$;
- if K is a multi-level knowledge node composed of sub nodes $K_1, \ldots, K_n$ and alignments A_{ij} between K_i and K_j for $i, j \in [1, n]$ and trans is the function that associates to any correspondence of A_{ij} a DL-axiom (see Definition 3.3) and alignment-ontology-formula set $F_A(K) = \{f \mid f \in \mathsf{trans}(A_{ij})\}$.

Definition 3.5 (Alignment-ontology)**.** Let us consider a node $K = \langle\{K_i\}, \{A_{ij}\}\rangle$ for $i, j \in [1, n]$, K_i are local nodes and A_{ij} is an alignment between K_i and K_j. We define $\mathsf{OntoAlign}$ the alignment-ontology generated from A_{ij} of K, $\mathsf{OntoAlign}(K) = \langle\Sigma_A(K), F_A(K)\rangle$.

The bridge rules of multi-level knowledge node represent the equivalence correspondences established between the terms of alignment-ontology and terms belonging to the corresponding local ontologies.

Definition 3.6 (Bridge rules toward alignment-ontology)**.** Let us consider a knowledge node K. The case dependant, bridge rules oriented towards the alignment-ontology (noted $B(K)$) is defined as follows:

- if K is an ontology then $B(K) = \emptyset$;
- if K a multi-level knowledge node composed of sub nodes $K_1, \ldots, K_n$ and A_{ij} which is alignment between K_i and K_j for $i, j \in [1, n]$ then $B(K)$ contains a bridge rules defined as follows, for $i \in [1, n]$:
  - if K_i is an ontology and X is a concept or a role of K_i then $i{:}X \overset{\equiv}{\rightarrow} \mathsf{OntoAlign}(K){:}i{:}X \in B(K)$;
  - if K_i is an ontology a is an individual of K_i then $i{:}a \overset{\equiv}{\rightarrow} \mathsf{OntoAlign}(K){:}i{:}a \in B(K)$;
  - if K_i is a composed node and X a concept or role of $\mathsf{OntoAlign}(K_i)$ then $\mathsf{OntoAlign}(K_i){:}X \overset{\equiv}{\rightarrow} \mathsf{OntoAlign}(K){:}k_i{:}X \in B(K)$;
  - if K_i is a composed node and a an individual of $\mathsf{OntoAlign}(K_i)$ then $\mathsf{OntoAlign}(K_i){:}a \overset{\equiv}{\rightarrow} \mathsf{OntoAlign}(K){:}k_i{:}a \in B(K)$.

The MLNK interpreted as a DDL system is composed of several local nodes connected to their alignment-ontology through a family on bridge rules.

Definition 3.7 (MLNK in DDL form)**.** Let us consider a knowledge node K. $\mathsf{SystDis}$ is a DDL system of K, $\mathsf{SystDis}(K) = \langle\mathsf{Onto}(K), \mathsf{Bridge}(K)\rangle$ with $\mathsf{Onto}(K)$ a family of local ontologies which is recursively defined as follows

- $\mathsf{Onto}(K) = \{K\}$, if K is a DL-ontology;
- $\mathsf{Onto}(K) = \mathsf{Onto}(K_1) \cup \mathsf{Onto}(K_2) \cup \ldots \cup \mathsf{Onto}(K_n) \cup \mathsf{OntoAlign}(K)$ if K is a node with K_i local nodes.

$\mathsf{Bridge}(K)$ is a family of bridge rules of K recursively defined as follows:

- $\mathsf{Bridge}(K) = \emptyset$ if K is an ontology;
- $\mathsf{Bridge}(K) = \mathsf{Bridge}(K_1) \cup \ldots \cup \mathsf{Bridge}(K_n) \cup B(K)$.

We will illustrate this transformation with examples:

Example 2. Let us consider a networked ontologies $K = \langle \{O_1, O_2\}, \{A_{12}\} \rangle$, with $A_{12} = \{1{:}A \xleftrightarrow{\sqsubseteq} 2{:}B, 1{:}a \xleftrightarrow{L} 2{:}b\}$ where A, B are concepts or roles, a, b are individuals and L is a *link*. We can say that an interpretation $\mathcal{I}$ satisfies K if $\mathcal{I}$ satisfies O_1 and O_2 and it also satisfies A_{12}. To interpret K according to the DDL-approach, we transform it into a distributed system $\mathsf{SystDis}(K) = \langle \{O_1, O_2, O_3\}, \{b_1, b_2, b_3, b_4\} \rangle$ with $O1$, $O2$ being the source ontologies, O_3 is an alignment-ontology generated from the alignments and b_1, b_2, b_3, b_4 are equivalence bridge rules.

- $b_1 = 1{:}A \stackrel{\equiv}{\rightarrow} 3{:}1{:}A$;
- $b_2 = 1{:}a \stackrel{\equiv}{\rightarrow} 3{:}1{:}a$;
- $b_3 = 2{:}B \stackrel{\equiv}{\rightarrow} 3{:}2{:}B$;
- $b_4 = 2{:}b \stackrel{\equiv}{\rightarrow} 3{:}2{:}b$;

b_1, b_2, b_3, b_4 are interpreted by the domain relations that bind the corresponding local interpretations according to the DDL semantics:

- $\mathcal{I}_1, \mathcal{I}_3 \models b_1$ if $r_{13}(A^{\mathcal{I}_1}) = 1 : A^{\mathcal{I}_3}$;
- $\mathcal{I}_1, \mathcal{I}_3 \models b_2$ if $r_{13}(a^{\mathcal{I}_1}) = 1 : a^{\mathcal{I}_3}$;
- $\mathcal{I}_2, \mathcal{I}_3 \models b_3$ if $r_{23}(B^{\mathcal{I}_2}) = 1 : B^{\mathcal{I}_3}$;
- $\mathcal{I}_2, \mathcal{I}_3 \models b_4$ if $r_{23}(b^{\mathcal{I}_2}) = 1 : b^{\mathcal{I}_3}$;

The interpretation K satisfies the correspondences of K if :

- $\mathcal{I} \models 1{:}A \xleftrightarrow{\sqsubseteq} 2{:}B$ if $\mathcal{I}_1, \mathcal{I}_3 \models b_1$ and $(1 : A)^{I_3} \subseteq (2 : B)^{I_3}$ and $\mathcal{I}_2, \mathcal{I}_3 \models b_3$;
- $\mathcal{I} \models 1{:}a \xleftrightarrow{L} 2{:}b$ if $\mathcal{I}_1, \mathcal{I}_3 \models b_2$ and $L(1 : a^{I_3}, 2 : b^{I_3}) \in L^{\mathcal{I}_3}$ and $\mathcal{I}_2, \mathcal{I}_3 \models b_4$.

K distributed interpretation, $\mathcal{I} = \{\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, r_{13}, r_{23}\}$ where $\mathcal{I}_1, \mathcal{I}_2$ are the local interpretations of O_1, O_2, $\mathcal{I}_3$ is the interpretation of generated alignment-ontology and r_{13}, r_{23} are domain relations for interpreting generated rule bridges.

$\mathcal{I}$ satisfies the ontologies network K in the DDL-approach if $\mathcal{I}$ satisfies $\mathsf{SystDis}(K) = \langle \{O_1, O_2, O_3\}, \{b_1, b_2, b_3, b_4\} \rangle$ in the basic semantics DDL.

Example 3. Ontologies and alignments of Example 1 are used to build a DDL system. Table 2 details the contents of those nodes.

4 DDLMLNKR PROTOTYPE

The DDLMLNKR prototype exploits the distributed reasoner DRAGO [30], that can handle OWL ontologies and RDF/XML files containing *mappings* and *links* as inputs.

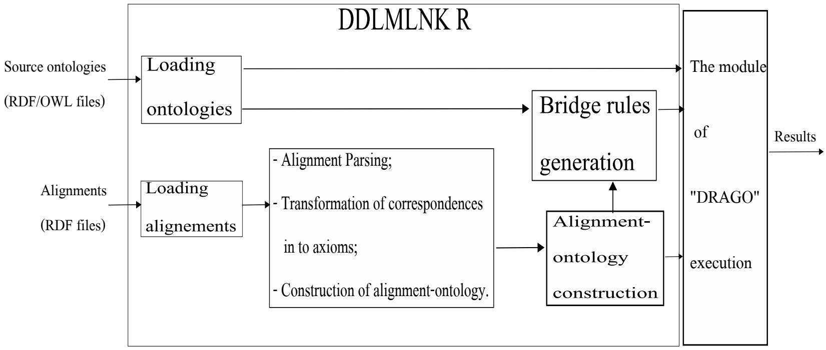

Fig. 7. DDLMLNKR architecture

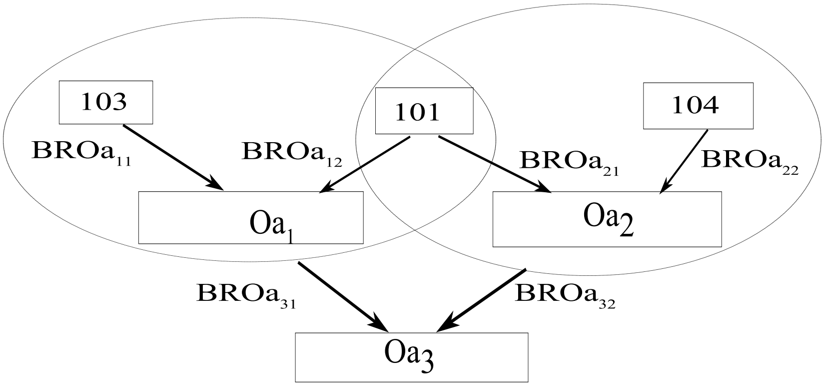

Fig. 8. MLNK transformation into a Distributed System

4.1 DDLMLNKR prototype architecture

The main components of this tool are illustrated in Figure 7 which describes the general architecture of the DDL-approach implementation. Each component is then described as follows:

- **Alignments loading**: It allows to loading alignments saved in RDF files, resulting from alignment discovery tools available on the World WideWeb. Alignment may be enriched in a semi-automatic manner using links;
- **Parser**: it allows parsing RDF/XML files containing alignments, it also allows recognizing *mappings* which are converted into axioms and *links* converted into specific roles;
- **Alignment-ontology generating**: in this module, the construction of an ontology in DL whose entities appear to the left and right of the alignment-

correspondences is performed. This module also integrates the axioms produced from the transformation of *mappings* and roles from *links*;

- **Bridges rules generating**: This component is used to generate the bridge rules between the entities belonging to the local ontologies and the corresponding entities belonging to the alignment-ontologies. They are then stored as C-OWL [9] files. C-OWL (Contextualized OWL) is an extension of OWL language designed to express *mappings* in DDL [8] formalism;
- **Executing module of distributed reasoner DRAGO**: URLs of the target ontology (alignment-ontology) and bridges rules are introduced and the source ontologies are determined by DRAGO. Subsequently, it will then be possible to determine the consistency of the networked ontologies.

Algorithm 4.1 Transformation of MLNK into a Distributed System

```
load($\{A_{ij}\}$) //$i, j \in [1, ..n]$
for all $A_{ij} \in \{A_{ij}\}$ do
    create( $O_{ak}, BRO_{ak1}, BRO_{ak2}$) //$k \in [1, ..n]$
    for all correspondence $c \in A_{ij}$ do
        read $i$:entity1, $j$:entity2
        $O_{ak}$.add($O_{ak}$:$i$:entity1)
        $O_{ak}$.add($O_{ak}$:$j$:entity2)
        if $c = map$ then
            transform $c$ into axiom
            $O_{ak}$.add($O_{ak}$:axiom)
        else
            transform $c$ into Object-property //($c = link$)
            $O_{ak}$.add($O_{ak}$:ObjectProperty)
        end if
        create equiv-map between $i$:entity1 and $O_{ak}$:$i$:entity1
        $BRO_{ak1}$.add(equiv-map)
        create equiv-map between $j$:entity2 and $O_{ak}$:$j$:entity2
        $BRO_{ak2}$.add(equiv-map)
    end for
end for
```

4.2 Implementation and experimentation of DDLMLNKR prototype

Experimentation tests were performed on Benchmark ontologies [1]. Table **??**, describes the size of the used ontologies and alignments constituting the MLNK. Inter-Ontology alignments $A_{101-103}, A_{101-104}$ were enriched by new *links* as they

---

[1] `http://oaei.ontologymatching.org/2014/`

did not contain any vocabulary. Then an Alignment, $A_{101-103-101-104}$ is created between inter-ontology alignments $A_{101-103}, A_{101-104}$, enriched by *mappings* between the *links* existing in the alignments $A_{101-103}, A_{101-104}$. Having the "Alignment API" format [13] extended earlier, in order to store *links*. A part of $A_{101-103-101-104}$ alignments is shown in Listing 1. A *mapping* representing an equivalence relation is inserted between the *links* "evaluate" and "reviewed"
Considering we have a MLNK, with existing alignments at several levels, $K = \langle\{101, 103, 104\}, \{A_{101-103}, A_{101-104}, A_{101-103-101-104}\}\rangle$. The transformation of the network to a distributed system SystDis(K) consists in generating (see Algorithm 4.1):

- Ontologies O_{a1}, O_{a2}, O_{a3} for the respective alignments $A_{101-103}, A_{101-104}$, $A_{101-103-101-104}$.
- Equivalence Bridge Rules between generated ontology alignments and source ontologies: $BRO_{a11}, BRO_{a12}, BRO_{a21}, BRO_{a22}, BRO_{a31}, BRO_{a32}$

$\mathsf{SystDis}(K) = \langle\{101, 103, 104, O_{a1}, O_{a2}, O_{a3}, BRO_{a11}, BRO_{a12}, BRO_{a21}, BRO_{a22}, BRO_{a31}, BRO_{a32}\}\rangle$ is the distributed obtained system. The transformation is depicted in Figure 8). The steps implemented during the transformation of the network, K, following the DDLMLNKR prototype are:

1. Load alignments $A_{101-103}, A_{101-104}$, $A_{101-103-101-104}$;
2. the prototype parses the alignments, identify correspondences of *mapping* types and transform them into axioms. The correspondences of *link* types are transformed into roles.
3. The prototype generates alignment-ontologies O_{a1}, O_{a2}, O_{a3}, having as a signature, entities being on the left and right of correspondences and roles resulting from *links* transformation. O_{a1}, O_{a2}, O_{a3} contains also, axioms resulting from correspondences transformation;
4. The prototype generates bridge rules $BRO_{a11}, BRO_{a12}, BRO_{a21}, BRO_{a22}, BRO_{a31}, BRO_{a32}$. This step consists in creating the correspondences of *mapping* type between entities in alignment-ontologies and their images in source ontologies.
5. The execution of the DRAGO reasoner for consistency test, is handled as follow:
    (a) Construction of the first $Peer1$ inserting the target ontology O_{a1} and Bridge rules BRO_{a11}, BRO_{a12}. For each bridge rule, the source ontology is identified and automatically inserted. As an example, for BRO_{a11} ontology 101 is inserted, as for BRO_{a12} it is ontology 103;
    (b) The second $Peer2$ is constructed by inserting the target ontology O_{a2} and bridge rules BRO_{a21}, BRO_{a22}. Source ontologies 101 and 104 are identified and inserted automatically;
    (c) The third $Peer3$ is constructed by inserting the target ontology O_{a3} and bridge rules BRO_{a31}, BRO_{a32}, Source ontologies O_{a1} and O_{a2} are identified and inserted automatically;
    (d) Run the consistency test for each $Peer$

A *Peer* is a concept of the DRAGO reasoner [30] consisting in regrouping for each target ontology, its own mappings as well as associated ontologies.

The distributed system, SystDis(K), is consistent if and only if: the Peer1, Peer2 and Peer3 are consistent. Results with respect to the transformation time and consistency time for the Network K, are presented in Section 5.4 for comparative analysis with DLMLNKR results presented in the paper [25].

Listing 1. A part of alignment ($A_{101-103-101-104}$)

```
<Alignment>
<alignment IRI = "http://.../alignment-101-103-101-104.rdf”/>
 <xml>yes</xml>
 <level>0</level>
  <type>11</type>
  <onto1>http://.../alignment-101-103.rdf</onto1>
  <onto2>http://.../alignment-101-104.rdf</onto2>
  <map>
  <Cell>
  <entity1 rdf:resource=
            'http://.../alignment-101-103#evaluate'/>
  <entity2 rdf:resource=
            'http://.../alignment-101-104#reviewed'/>
<measure rdf:datatype='http://...#float'>1.0</measure>
      <relation>=</relation>
</Cell>
</map>
```

5 DL-APPROACH AND DDL-APPROACH COMPARISON

In this section, DL and DDL-approaches are compared, with respect to specific criteria in order to determine for which cases one is more suitable than the other.
The two approaches are then studied with respect to both evaluation criteria and comparative summary tables are presented in Tables 4 and 5.

5.1 Consistency comparison

For consistency, the goal is to try to prove that an inconsistent multi-level networked knowledge expressed in the DL-approach, could be consistent in the DDL-approach.

Theorem 1. If a Multi-Level Networked knowledge is inconsistent when expressed in DL-approach, it can be consistent when expressed in DDL-approach.

This theorem can be proved by showing that the multi-level networked knowledge, in the example is inconsistent according to DL-approach semantics (ENCACS) and is consistent according to DDL-approach semantics (DACOSLS).

Example 4. Let us consider an ontologies $O_1 = \{A_1 \sqsubseteq \neg B_1, A_1(a)\}$, $O_2 = \{A_2 \sqsubseteq B_2\}$ and an alignment $A_{12} = \{1{:}A_1 \xleftrightarrow{\equiv} 2{:}A_2, 1{:}B_1 \xleftrightarrow{\equiv} 2{:}B_2\}$.

Lemma 1. DL-approach consistency: Constitute a global ontology whose elements are prefixed from source ontologies and the *mappings*, *links* are transformed into axioms $O_G = \{1{:}A_1 \sqsubseteq \neg 1{:}B_1, 1{:}A_1(1{:}a), 2{:}A_2 \sqsubseteq 2{:}B_2, 1{:}A_1 \equiv 2{:}A_2, 1{:}B_1 \equiv 2{:}B_2\}$.

$$1{:}A_1 \sqsubseteq \neg 1{:}B_1 \tag{1}$$

$$1{:}A_1(1{:}a) \tag{2}$$

$$2{:}A_2 \sqsubseteq 2{:}B_2 \tag{3}$$

$$1{:}A_1 \equiv 2{:}A_2 \tag{4}$$

$$1{:}B_1 \equiv 2{:}B_2 \tag{5}$$

$$1, 4, 5 \Rightarrow 2{:}A_2 \sqsubseteq \neg 2{:}B_2 \tag{6}$$

$$2, 4 \Rightarrow 2{:}A_2(1{:}a) \tag{7}$$

$$6, 7 \Rightarrow \neg 2{:}B_2(1{:}a) \tag{8}$$

$$7, 3 \Rightarrow 2{:}B_2(1{:}a) \tag{9}$$

Contradiction according to (8) and (9) and this implies that O_G is DL-approach inconsistent.

Lemma 2. DDL-approach consistency: Let us take the same Example 4, construct a distributed system S according to the DDL-approach, with an alignment-ontology constructed from the correspondences, noted O_{12}, generating then, the corresponding bridges rules $\mathcal{B}$.
We obtain an ontology $O_{12} = \{1{:}A_1 \equiv 2{:}A_2, 1{:}B_1 \equiv 2{:}B_2\}$ and the bridges rules
$\mathcal{B} = \{b_1, b_2, b_3, b_4\}$ where: $b_1 = \{1{:}A_1 \overset{\equiv}{\rightarrow} 12{:}(1{:}A_1)\}$; $b_2 = \{1{:}B_1 \overset{\equiv}{\rightarrow} 12{:}(1{:}B_1)\}$; $b_3 = \{2{:}A_2 \overset{\equiv}{\rightarrow} 12{:}(2{:}A_2)\}$; $b_4 = \{2{:}B_2 \overset{\equiv}{\rightarrow} 12{:}(2{:}B_2)\}$.

To show that $S = \{O_1, O_2, O_{12}, \mathcal{B}\}$ is consistent, then we must find a model that satisfies all axioms and bridges rules of S.
Supposing that a model of S exists then there is a distributed interpretation
$I = \{I_1, I_2, I_3, r_{12}, r_{13}, r_{23}, r_{21}, r_{31}, r_{32}\}$ such that $I \models S$.
This implies that: $I_1 \models O_1$; $I_2 \models O_2$; $I_3 \models O_{12}$; $I_1, I_3, r_{13} \models b_1$; $I_1, I_3, r_{13} \models b_2$; $I_2, I_3, r_{23} \models b_3$; $I_2, I_3, r_{23} \models b_4$;
This is equivalent to showing that there exists an interpretation $I = \{I_1, I_2, I_3, r_{13}, r_{23}\}$ such that: $A_1^{I_1} \subseteq \neg B_1^{I_1} \subseteq \Delta_1$; $a^{I_1} \in A^{I_1}$; $A_2^{I_2} \subseteq B_2^{I_2} \subseteq \Delta_2$; $r_{13}(A_1^{I_1}) = (1{:}A_1)^{I_3}$; $r_{13}(B_1^{I_1}) = (1{:}B_1)^{I_3}$; $r_{23}(A_2^{I_2}) = (2{:}A_2)^{I_3}$; $r_{23}(B_2^{I_2}) = (2{:}B_2)^{I_3}$.
Consider the following domain of interpretation: $\Delta_1 = \{1\}$, $\Delta_2 = \{2\}$, $\Delta_3 = \{3\}$, and interpretation functions defined as follows : $A_1^{I_1} = \{1\}$; $a^{I_1} = 1$; $B_1^{I_1} = \emptyset$; $A_2^{I_2} = \emptyset$; $B_2^{I_2} = \emptyset$; $(1{:}A_1)^{I_3} = \emptyset$; $(1{:}B_1)^{I_1} = \emptyset$; $(2{:}A_2)^{I_3} = \emptyset$; $(2{:}B_2^{I_3}) = \emptyset$; $r_{13}(A_1^{I_1}) = \emptyset$; $r_{13}(B_1^{I_1}) = \emptyset$;

$r_{23}(A_2^{I_2}) = \emptyset$; $r_{23}(B_2^{I_2}) = \emptyset$.
So for $I = \langle(\{1\}, I_1), (\{2\}, I_2), (\{3\}, I_3), \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset\rangle$, we have $I \models S$, then S is consistent.

It can then be concluded that the way the alignments are treated when expressed in DL, is fixed and thus allows a reconciliation of local ontologies and alignments. This approach can be used in the case of modular ontologies, alignments, where each module is part of a global perspective in a broader domain. However, It has limits when it comes to the Word Wide Web, where ontologies, alignments can have contradictory points of view. On the other hand, when expressed in DDL, ontologies with different viewpoints may collaborate, even if they are considered incompatible.

5.2 Transformation complexity

- **The complexity of transforming the multi-level networked knowledge into a DL-ontology is linear in terms of ontologies and corresponding alignments (*comptDL*). It can be calculated using the number of prefix (ontology and links terms), noted (*nbprefix*) and the generated axiom number noted (*nbaxiom*). Let the variables n_i, l, m and p, be respectively the number of local terms belonging to the local ontology O_i, the number of links term, the number of levels and the number of correspondences**;

$$nbprefix = (m-1) * l + m * \sum n_i \tag{10}$$
$$nbaxiom = p \tag{11}$$
$$comptDL = nbprefix + p \tag{12}$$

- The transformation complexity in a DDL distributed system (comptDDL) is calculated according to the number of operations performed to create axioms in the alignment ontology (Axioms are obtained from the transformation of correspondences), and the number of bridge rules creation operations (nbbr). Let us recall that for a correspondence there are two terms (the terms on the right and the terms on the left of the correspondence) and for each term, a bridge rule is created;

$$nbaxioma = p \tag{13}$$
$$nbbr = 2p \tag{14}$$
$$comptDDL = nbaxioma + nbbr \tag{15}$$

The transformation complexity in the case of updating local ontologies expressed in DL is proportional to the number of updates, bearing in mind that, updating local ontologies leads to the reconstruction of a global ontology. For the DDL- approach, the update of the local ontologies does not affect the transformation. Thus, it can be concluded that DDL-approach is more appropriate in the case where the evolution of local ontologies is more important than that of the correspondences.

5.3 Reasoning complexity

Reasoning complexity of MLNK semantic-approaches is based on the reasoning complexities of the basic semantics of DL and DDL. Multi-Level Networked Knowledge in DL-approach is transformed into DL-ontology constructed from a fusion of local ontologies whose terms have been prefixed and alignments transformed into axioms. The local ontologies can be formalized in different logics, with the expressivity of the axiom' origin alignment being very simple and possibly formalized in the decidable $\mathcal{EL}$ language whose complexity is *NPcomplete*. Thus, the decidability and the complexity of the MLNK interpreted in DL can be given by studying the decidability and the complexity fusion of the local description logics and the integrated axioms logics. In that context, a recent work addressing the reasoning complexity in multi-viewpoint ontologies, via import from other ontologies may be of interest [23].

This aspect has not been dealt with in this paper, however, the reader is redirected to [6] for a more comprehensive description. First, this work shows that the fusion of two description logic is a fragment of the union of the latter because reasoning on the union of the two logics requires the implementation of a new reasoning method. However, reasoning on the merger can be reduced to reasoning on logical components. Moreover, reasoning on the union of two decidable logics can be undecidable, whereas reasoning on the fusion of the same logic remains decidable.

For example, the union of logics $\mathcal{ALCF}$ (which is an extension of $\mathcal{ALC}$ by the addition of functional roles) and $\mathcal{ALC}^{+,\circ,\cup}$ (Which is an extension of $\mathcal{ALC}$ by the addition of transitivity, composition and union of roles), is undecidable. While their fusion is decidable. According to the same paper, the complexity of the description logics merge, whose complexity is **Pspace** is also **Pspace** [6]. This is not valid for the union of these logics. For example, the complexity of the union of logics $\mathcal{ALCFOQ}$ (which is an extension of $\mathcal{ALC}$ by adding functional role, nominal and number restriction) and the $\mathcal{ALCI}$ logic (which is an extension of $\mathcal{ALC}$ by the addition of inverse role) is **NExpTime** whereas the complexity of the component logic is **PSpace** [6]. This is different for the DDL-approach, where the logics are not merged but connected by relationships, Ghidini and al. in [18] present a study showing that the inference on *mappings* is decidable and the complexity ranges between **ExpTime** and **2ExpTime**. It can then be concluded, that the complexity of the MLNK interpreted in DDL can be equal to the highest complexity among local ontologies and mappings inferences.

5.4 Comparison of MLNK prototypes

The results are given by the MLNK transformation test performed by the two prototypes DLMLNKR and DDLMLNKR on the initial ontologies (Case 1) show that the transformation time in a distributed system is slightly improved over the one obtained constructing a global DL ontology, see Table 4 and Figure 9 (Case 1).
Case 2 evaluates the impact of the source ontology evolution on transformation time. Ontologies have been enriched by new entities, independent from alignments. This permits to enlarge the source ontology sizes, keeping the alignment size unchanged. Then results presented in Table 4 and Figure 9 (Case 2) show that the transformation time of the MLNK using the DDLMLNKR prototype remain unchanged. This concludes that the DDL-approach is transparent with respect to ontology evolution.
In Case 3, the impact of alignment evolution is tested, with the insertion of *mappings* and *links* performed between existing entities. The goal is to increase alignments size while keeping ontology size unchanged. The results show that transformation results using both prototypes are affected. This concludes that MLNK transformation time evolves

with respect to the evolution of alignments size. Table 4 and Figure 9 (case 3) show that reasoning upon distributed semantic is context depending, and more computationally expensive than reasoning based on a non contextual one. However, according to Section 5.1, it has been proven that the consistency test for contextual semantics is more efficient than that of not-contextual semantics. Let us suppose that for a given case, the consistency test following a DL-approach is inconsistent and that entities causing the inconsistency belong to different ontologies, however, not concerned by alignments. In that case, the network is consistent following the DDL-approach.
Based on the consistency test for all three studied cases, it is clear that the evolution of ontology and alignment sizes does not affect consistency at all. In other words, evolution does not affect complexity, (Table 4 and Figure 10).

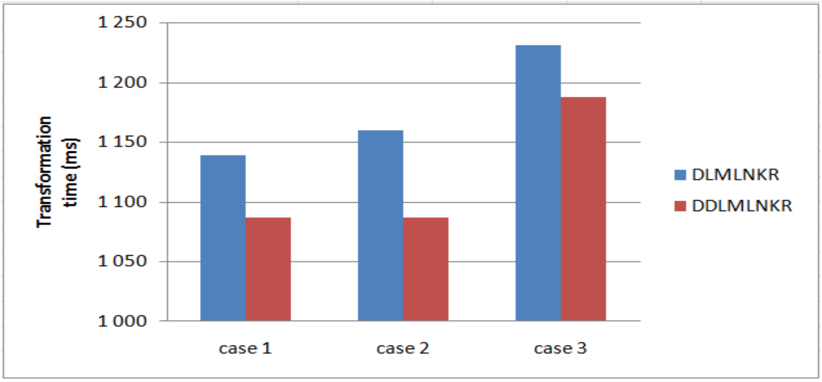

Fig. 9. MLNK transformation test results

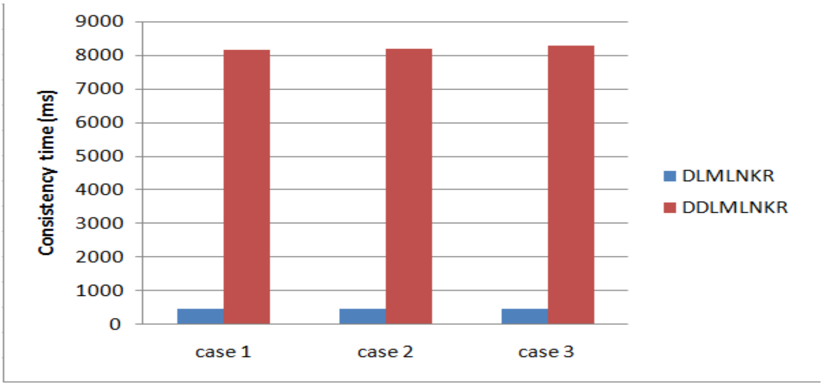

Fig. 10. MLNK consistency test results

6 STATE OF ART SYNTHESIS AND DISCUSSION

In previous works, authors have surveyed research in relation to the topic of MLNK formalisms [25] and [24], and do not wish to develop them further again in the present work, stating only the most recent ones. Previous research have been classified into two main research categories: "aligned knowledge networks" and "contextual knowledge modelling". In the first category "aligned knowledge networks", research focus on representation and reasoning on heterogeneous ontologies built independently however still aligned. This is the case in Distributed Description Logic [8], Integrated Distributed Description Logics [33], Package-based Description Logics [7], $\mathcal{E}$-connection [27] and $E-\mathcal{SHIQ}$ [32], as well as the proposed formalism. Works classified in the second category "contextual knowledge modelling", model the contexts, linking those via a meta description. Each context posses then its own instances and use aggregation relations in order to link instances. As examples, [26], [22], [21], and recently [4] as well as [19], fall into this category, with the latter reference proposing reasoning on a hierarchical structure of the contexts.

The difference between the categories vision "contextual knowledge modelling" and "aligned knowledge networks", is similar to the difference between the Global-As-View (GAV) and Local-As-View (LAV) approaches used in integration data systems formalized and expressed in terms of requests [10], [15].

The modelling principle of works in "contextual knowledge modelling" category is the same as that of GAV where a top-down design approach is applied, proceeding from global to local. On the other hand, for the works in "aligned knowledge networks" category and LAV approaches, the upward design method is applied from local to global.

Other works, consider that every local source in a network is treated as an independent module, permitting reasoning on the latter [20] and [29].

In this paper, stress is put on formalisms that represent and reason on independent and aligned ontologies. Differences between presented formalisms will be discussed, with a special attention given to the contribution of the proposed formalism. A summary of the above is depicted in Table 6.

6.1 Multi-level networked knowledge representation

Multi-level networked knowledge is composed of a set of aligned nodes, these in turn are composed of the aligned sub-nodes and so on, where the most elementary nodes are ontologies. The alignment of the nodes composed of sub-nodes and alignments between them makes it possible to align the alignments and thanks to this structure the alignments can be formalized. No formalism cited below tolerates a dynamic representation of local and aligned knowledge. In addition, the syntactic formalization of local knowledge (ontology and nodes) in the proposed formalism is described in an abstract and independent way from any language and consequently, can be adapted to any logic. DDL [8], P-DL [7], $E-\mathcal{SHIQ}$ and IDDL [33] are developed for a network of description logic ontologies. The ontologies in DFOL [17] formalism can be expressed in first-order logic. In $\mathcal{E}$-connection [27], the local ontologies of the same network can be represented in various logics along with an abstract description system.

6.2 Alignment contextual representation

In multi-level networked knowledge, alignments are expressed using an alignment language independently from ontology languages. These have their own vocabularies, consisting in *mappings* and/or *links* and expressed according to the point of view of the pair of ontologies combination. In other words, according to the global point of view in relation to a pair of ontologies. Unlike DDL and IDDL that only define and interpret *mappings*, $\mathcal{E}$-connection [27] and $E - \mathcal{SHIQ}$ [32] express *links* but do not take into account the conflict of alignment heterogeneity. This is mainly because they are oriented and interpreted according to the target ontology correspondence point of view. The definition of the correspondences for a global point of view has already been presented in the IDDL formalism, but given the absence of *links* (therefore of alignment vocabulary), it does not require alignment of higher levels.

6.3 The semantics associated with multi-level networked knowledge formalism

For interpretation, an instantiation of the generic formalism is carried out. We are interested in the case where ontologies are expressed in description logics (DL).

- The DL-approach that adopts ENCACS, the basic Non-Contextual And Centralized Semantics is applied by SomeWhere [1] and SomeRDFS [3], SomeOWL [2] and OWL's import semantics;
- The DDL-approach adopts Distributed and Contextual-on-several-levels Semantics, the basic distributed and contextual semantics is applied using DDL, PDL, $\mathcal{E}$-connection and $E-\mathcal{SHIQ}$. In our case, the alignments are not interpreted according to the target ontology correspondence viewpoint, but they are interpreted in an external level. Independently of local ontologies, this external level is represented by an interpretation domain associated to generated alignment-ontology.

6.4 Reasoning

Several reasoning prototypes may be associated with MLNK. DLMLNKR prototype [25] allows reasoning on the proposed formalism adopting the DL-approach. The SomeWhere and SomeRDF algorithms can also be exploited, (but only when links are ignored) to ensure a distributed and not-contextual reasoning. The DDL-approach implementation (DDLMLNKR prototype, Section 4.2) is ensured using the DRAGO reasoner and allows a distributed and contextual reasoning on the MLNK.

7 CONCLUSIONS

This work is the extension of previous works [25] and [24], and proposes an extended semantics that can be associated with MLNK. The main advantage of those semantics is their ability to handle separately alignment interpretations. The DACOSLS is not only suitable for contextual ontology reasoning, but

also for contextual alignment reasoning. In order to prove the feasibility and efficiency of the DDL-approach which adopts DACOSLS, a prototype based on the DRAGO reasoner and termed DDLMLNKR was designed and implemented. Results on consistency tests and transformation time are assessed and commented, as well as compared to the ones obtained using the DL-approach. Based on the viewpoint notion, it can be concluded that DL-approach may be used in cases where interpretation domains of the network local sources are defined in different but compatible contexts. Each domain consists then in a portion of completing others in the larger domain. DDL-approach is therefore recommended in the case where local sources interpretation domains (Ontologies and Alignments) of the network are defined in different incompatible contexts, thus permitting contextualization of ontologies and alignments. Other comparison criteria, may be useful to help users choose the most appropriate approach for their applications.

However, the introduction of such structures poses new practical and theoretical issues, which we would like to explore later may be given by:

1. One can wonder about the problem of automatic correspondences discovery between alignments: are the tools and techniques used for ontology alignment construction adapted to all levels of a knowledge network? Can alignments be used at a certain level for the discovery of higher level alignments?
2. The need for a concise representation of such networks in a possible standardized format;
3. Knowledge management or visualization tools need to be built to organize and observe multi-level networks in order to maintain them throughout their life cycle. In addition, the hierarchical construction of multi-level networks requires re-evaluating knowledge modelling methodologies by detailing the steps to be followed for their development;
4. Concerning the semantic part, the use of existing paradigms was privileged. However, It would be interesting to reflect on another way of interpreting the MLNK semantics by defining a formal semantics constructed directly on this structure and then propose a correct and complete reasoning algorithm;
5. Finally, it would be important and useful to develop a system able of interrogating this type of network. A formalization of the federated request system is under development and will be presented later.

8 *

REFERENCES

[1] Philippe Adjiman, Philippe Chatalic, François Goasdoué, Marie-Christine Rousset, and Laurent Simon. SomeWhere in the Semantic Web. In François Fages and Sylvain Soliman, editors, *Principles and Practice of Semantic Web Reasoning, Third International Workshop, PPSWR 2005, Dagstuhl Castle, Germany, September 11-16, 2005, Proceedings*, volume 3703 of *Lecture Notes in Computer Science*, pages 1–16. Springer-Verlag, 2005.

[2] Philippe Adjiman, Philippe Chatalic, François Goasdoué, Marie-Christine Rousset, and Laurent Simon. Distributed Reasoning in a Peer-to-Peer Setting: Application to the Semantic Web. *Journal of Artificial Intelligence Research*, 25:269–314, 2006.

[3] Philippe Adjiman, François Goasdoué, and Marie-Christine Rousset. SomeRDFS in the Semantic Web. *Journal on Data Semantics*, 8:158–181, 2007.

[4] Sahar Aljalbout and Gilles Flaquet. Practical Implementation of Contextual Reasoning on the Semantic Web. Springer-Verlag, 2018.

[5] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications.* Cambridge University Press, 2003.

[6] Franz Baader, Carsten Lutz, Holger Sturm, and Frank Wolter. Fusions of Description Logics and Abstract Description Systems. *Journal of Artificial Intelligence Research*, 16:1–58, 2002.

[7] Jie Bao, George Voutsadakis, Giora Slutzki, and Vasant G. Honavar. Package-Based Description Logics. In Heiner Stuckenschmidt, Christine Parent, and Stefano Spaccapietra, editors, *Modular Ontologies: Concepts, Theories and Techniques for Knowledge Modularization*, volume 5445 of *Lecture Notes in Computer Science*, pages 349–371. Springer-Verlag, 2009.

[8] Alex Borgida and Luciano Serafini. Distributed Description Logics: Assimilating Information from Peer Sources. *Journal on Data Semantics*, 1:153–184, 2003.

[9] Paolo Bouquet, Fausto Giunchiglia, Frank van Harmelen, Luciano Serafini, and Heiner Stuckenschmidt. C-OWL: Contextualizing Ontologies. In Dieter Fensel, Katia P. Sycara, and John Mylopoulos, editors, *The Semantic Web - ISWC 2003, Second International Semantic Web Conference, Sanibel Island, FL, USA, October 20-23, 2003, Proceedings*, volume 2870 of *Lecture Notes in Computer Science*, pages 164–179. Springer-Verlag, October 20 2003.

[10] Andrea Cali, Diego Calvanese, Giuseppe de Giacomo, and Maurizio Lenzerini. On the expressive power of data integration systems. In : International Conference on Conceptual Modeling. Springer, Berlin, Heidelberg. pages 338–350, 2002.

[11] Diego Calvanese, Enrico Franconi, Volker Haarslev, Domenico Lembo, Boris Motik, Sergio Tessaris, and Anni-Yasmin Turhan, editors. *Proceedings of the 20th International Workshop on Description Logics DL'07, June 8 - 10, 2007, Brixen/Bressanone, Italy.* Bolzano University Press, June 2007.

[12] Chan Le Duc, Myriam Lamolle, Antoine Zimmermann, and Olivier Curé. DRAOn: A Distributed Reasoner for Aligned Ontologies. In Samantha Bail, Birte Glimm, Rafael Gonçalves, Ernesto Jiménez-Ruiz, Yevgeny Kazakov, Nicolas Matentzoglu, and Bijan Parsia, editors, *Informal Proceedings of the 2nd International Workshop on OWL Reasoner Evaluation (ORE-2013)*, volume 1015 of *CEUR Workshop Proceedings*, pages 81–86. Sun SITE Central Europe (CEUR), July 2013.

[13] Jérôme Euzenat. An API for Ontology Alignment. In Frank van Harmelen, Sheila McIlraith, and Dimitri Plexousakis, editors, *The Semantic Web - ISWC 2004: Third International Semantic Web Conference,Hiroshima, Japan, November 7-11, 2004. Proceedings*, volume 3298 of *Lecture Notes in Computer Science*, pages 698–712. Springer-Verlag, 2004.

[14] Jérôme Euzenat and Pavel Shvaiko. *Ontology Matching.* Springer-Verlag, Heidelberg (DE), 2007.

[15] Michel Frank. *Integrating heterogeneous data sources in the WEB of data.* PhD thesis, Université Côte dAzur, France, 2017.

[16] Chiara Ghidini and Fausto Giunchiglia. Local Models Semantics, or contextual reasoning=Locality+Compatibility. *Artificial Intelligence*, 127(2):221–259, 2001.

[17] Chiara Ghidini and Luciano Serafini. Distributed First Order Logics. In Dov M. Gabbay and Maarten de Rijke, editors, *Frontiers of Combining Systems 2*, volume 7 of *Studies in Logic and Computation*, pages 121–139. Research Studies Press, 2000.

[18] Chiara Ghidini, Luciano Serafini, and Sergio Tessaris. Complexity of reasoning with expressive ontology mappings. In *FOIS*, pages 151–163, 2008.

[19] Krzysztof Goczyla, Aleksander Waloszek, and Wojciech Waloszek. Contextualization of a DL Knowledge Base. In Calvanese et al. [11].

[20] Krzysztof Goczyla, Aleksander Waloszek, Wojciech Waloszek, and Terese Zawadzka. Theoretical and Architectural Framework for Contextual Modular Knowledge Bases. In *Intelligent Tools for Building a Scientific Information Platform. springer Berlin, Heidelberg.*, pages 257–280, 2013.

[21] Mathew Joseph. *Query Answering over Contextualized RDF/OWL rules: Decidable classes.* PhD thesis, University of Trento, 2015.

[22] Mathew Joseph and Luciano Serafini. Simple Reasoning for Contextualized RDF Knowledge. In Oliver Kutz and Thomas Schneider, editors, *Modular Ontologies - Proceedings of the Fifth International Workshop, WoMO 2011, Ljubljana, Slovenia, August 2011*, volume 230 of *Frontiers in Artificial Intelligence and Applications*, pages 79–93. IOS Press, August 2011.

[23] Yevgeny Kazakov and Denis Ponomaryov. On the Complexity of Semantic Integration of OWL Ontologies. arXiv preprint arXiv:1705.04719. 2017.

[24] Sihem Klai, Antoine Zimmermann, and Mohamed Tarek Khadir. *Multi-level Networked Knowledge Base: DDL reasoning*, volume 9893 of *Lecture Notes in Computer Science.* Springer-Verlag, 2016.

[25] Sihem Klai, Antoine Zimmermann, and Mohamed Tarek Khadir. Multi-level Networked Knowledge: Reapresentation and DL reasoning. *International Journal of Metadata, Semantics and ontologies*, 11(01):1–15, 2016.

[26] Szymon Klarman. *Reasoning with Contexts in Description Logics.* PhD thesis, Vrije Universiteit, Amsterdam (Netherland), January 2013.

[27] Oliver Kutz, Carsten Lutz, Frank Wolter, and Michael Zakharyaschev. $\mathcal{E}$-connections of abstract description systems. *Artificial Intelligence*, 156(1):1–73, 2004.

[28] John L. McCarthy. Notes on formalizing context. In *Proceeding of the 13th international joint conference on Artificial Intelligence, IJAI'93*, 1993.

[29] Adrian Paschke and Ralph Schafermeier. OntoMaven-Maven-based Ontology Development and Management of Distributed Ontology Repositories In Synergies Between Knowledge Engineering and Software Engineering, Springer, Cham, 2018.

[30] Luciano Serafini and Andrei Tamilin. DRAGO: Distributed Reasoning Architecture for the Semantic Web. In Asunción Gómez-Pérez and Jérôme Euzenat, editors, *The Semantic Web: Research and Applications, Second European Semantic Web Conference, ESWC 2005, Heraklion, Crete, Greece, May 29 - June 1, 2005, Proceedings*, volume 3532 of *Lecture Notes in Computer Science*, pages 361–376. Springer-Verlag, May 2005.

[31] Evren Sirin, Bijan Parsia, Bernardo Cuenca-Grau, Aditya Kalyanpur, and Yarden Katz. Pellet: A pratical OWL-DL reasoner. *Journal of Web Semantics*, 5(2):51–53, 2007.

[32] George Vouros and Georgios M. Santipantakis. Distributed Reasoning with $\mathrm{E}^{\mathrm{DDL}}{}_{\mathrm{HQ}^{+}}$ SHIQ. In Jodi Schneider and Dirk Walther, editors, *Proceedings of the 6th International Workshop on Modular Ontologies, Graz, Austria, July 24, 2012*, volume 875 of *CEUR Workshop Proceedings*. Sun SITE Central Europe (CEUR), July 2012.

[33] Antoine Zimmermann. Integrated Distributed Description Logics. In Calvanese et al. [11], pages 507–514.

[34] Antoine Zimmermann and Jérôme Euzenat. Three Semantics for Distributed Systems and their Relations with Alignment Composition. In Isabel F. Cruz, Stephan Decker, Dean Allemang, Christ Preist, Daniel Schwabe, Peter Mika, Michael Uschold, and Lora Aroyo, editors, *The Semantic Web - ISWC 2006, 5th International Semantic Web Conference, ISWC 2006, Athens, GA, USA, November 5-9, 2006, Proceedings*, volume 4273 of *Lecture Notes in Computer Science*, pages 16–29. Springer-Verlag, November 2006.

[35] Antoine Zimmermann and J. M. Gimnez-Garca. Integrating Context of Statements within Description Logics. In *arXiv preprint arXiv:1709.04970.*, 2017.

Node	Distributed system
level 0	
$K_1 = \mathtt{pr}$	$B(K_1) = \emptyset, \mathsf{Onto}(K_1) = \{K_1\}, \mathsf{Bridge}(K_1) = \emptyset, \mathsf{SystDis}(K_1) = \{\{K_1\}, \emptyset\}$
$K_2 = \mathtt{eq}$	$B(K_2) = \emptyset, \mathsf{Onto}(K_2) = \{K_2\}, \mathsf{Bridge}(K_2) = \emptyset, \mathsf{SystDis}(K_2) = \{\{K_2\}, \emptyset\}$
$K_3 = \mathtt{zn}$	$B(K_3) = \emptyset, \mathsf{Onto}(K_3) = \{K_3\}, \mathsf{Bridge}(K_3) = \emptyset, \mathsf{SystDis}(K_3) = \{\{K_3\}, \emptyset\}$
level 1	
$K_4 = \{K_1, K_2, A_{\mathtt{K}_1\text{-}\mathtt{K}_2}\}$	$\mathsf{OntoAlign}(K_4) = \mathsf{oa}_4 = \langle \Sigma_4, F_4 \rangle$, where $\Sigma_4 = \{k_1{:}\mathtt{G}_1, k_2{:}\mathtt{DF}_1, \mathbf{compose}\}$ and $F_4 = \{\mathbf{compose}(k_1{:}\mathtt{G}_1, k_2{:}\mathtt{DF}_1)\}$ $B(K_4) = \{k_1{:}\mathtt{G}_1 \overset{\equiv}{\rightarrow} \mathsf{oa}_4{:}k_1{:}\mathtt{G}_1,$ $k_2{:}\mathtt{DF}_1 \overset{\equiv}{\rightarrow} \mathsf{oa}_4{:}k_2{:}\mathtt{DF}_1\};$ $\mathsf{Onto}(K_4) = \{K_1, K_2, \mathsf{oa}_4\}$ $\mathsf{Bridge}(K_4) = B(K_4);$ $\mathsf{SystDis}(K_4) = \langle \mathsf{Onto}(K_4), \mathsf{Bridge}(K_4) \rangle$
$K_5 = \{K_2, K_3, A_{\mathtt{K}_2\text{-}\mathtt{K}_3}\}$	$\mathsf{OntoAlign}(K_5) = \mathsf{oa}_5 = \langle \Sigma_5, F_5 \rangle$, where $\Sigma_5 = \{k_2{:}\mathtt{DF}_1, k_3{:}\mathtt{ANNA1TG01}, \mathbf{part\text{-}of}\}$ and $F_5 = \{\mathbf{part\text{-}of}(k_2{:}\mathtt{DF}_1, k_3{:}\mathtt{ANNA1TG01})\}$ $B(K_5) = \{k_2{:}\mathtt{DF}_1 \overset{\equiv}{\rightarrow} \mathsf{oa}_5{:}k_2{:}\mathtt{DF}_1,$ $k_3{:}\mathtt{ANNA1TG01} \overset{\equiv}{\rightarrow} \mathsf{oa}_5{:}k_3{:}\mathtt{ANNA1TG01}\};$ $\mathsf{Onto}(K_5) = \{K_2, K_3, \mathsf{oa}_5\}$ $\mathsf{Bridge}(K_5) = B(K_5)$ $\mathsf{SystDis}(K_5) = \langle \mathsf{Onto}(K_5), \mathsf{Bridge}(K_5) \rangle$
level 2	
$K_6 = \{K_4, K_5, A_{\mathtt{K}_4\text{-}\mathtt{K}_5}\}$	$\mathsf{OntoAlign}(K_6) = \mathsf{oa}_6 = \langle \Sigma_6, F_6 \rangle$ where $\Sigma_6 = \{\mathsf{oa}_4{:}\mathbf{compose}, \mathsf{oa}_5{:}\mathbf{part\text{-}of}\}$ and $F_6 = \{\mathsf{oa}_4{:}\mathbf{compose} \equiv \mathsf{oa}_5{:}\mathbf{part\text{-}of}\}$ $B(K_6) = \{\mathsf{oa}_4{:}\mathbf{compose} \overset{\equiv}{\rightarrow} \mathsf{oa}_6{:}\mathbf{compose},$ $\mathsf{oa}_5{:}\mathbf{part\text{-}of} \overset{\equiv}{\rightarrow} \mathsf{oa}_6{:}\mathbf{part\text{-}of}$ $\mathsf{Onto}(K_6) = \mathsf{Onto}(K_4) \cup \mathsf{Onto}(K_5) \cup \{\mathsf{oa}_6\}$ $= \{K_1, K_2, K_3, \mathsf{oa}_4, \mathsf{oa}_5, \mathsf{oa}_6\}$ $\mathsf{Bridge}(K_6) = \mathsf{Bridge}(K_4) \cup \mathsf{Bridge}(K_5) \cup B(K_6)$ $= B(K_4) \cup B(K_5) \cup B(K_6)$ $\mathsf{SystDis}(K_6) = \langle \mathsf{Onto}(K_6), \mathsf{Bridge}(K_6) \rangle$

Table 2. Example of an MLNK in DDL form. We rename $\mathsf{OntoAlign}(K_i)$ in oa_i for $i \in [4, 6]$.

	Ontologies/alignments	Size(kB)	DLMLNKR Time(ms)	DDLMLNKR Time(ms)
Case 1	101	71.5	Transformation= 1140 Consistency= 460	Transformation= 1087 Consistency= 8200
	103	80.4		
	104	46.1		
	$A_{101-103}$	44.4		
	$A_{101-104}$	49.3		
	$A_{101-103-101-104}$	4.45		
Case 2	101	104.2	Transformation= 1161 Consistency= 464	Transformation= 1087 Consistency= 8222
	103	110.3		
	104	78.8		
	$A_{101-103}$	44.4		
	$A_{101-104}$	49.3		
	$A_{101-103-101-104}$	4.45		
Case 3	101	71.5	Transformation= 1232 Consistency= 477	Transformation= 1189 Consistency= 8302
	103	80.4		
	104	46.1		
	$A_{101-103}$	78.2		
	$A_{101-104}$	84.5		
	$A_{101-103-101-104}$	9.01		

Table 4. Comparison of MLNK prototypes results

Approach	point of view of ontologies and alignments	Impact of updating ontologies	Reasoner implementation	Transformation complexity	Reasoning complexity
DL-approach	compatible viewpoints	leads to the network updating	supports multiple logics	increases with increasing sizes of ontologies and alignments	= complexity of the local logics merge
DDL-approach	inconsistent viewpoints	transparent to the network	a reasoner by logic	increases with increasing sizes of alignments	= the highest among the complexities of local logics or mappings inference

Table 5. Comparative table of DL and DDL-approaches

Formalism: DDL
Motivation: resolution of semantic heterogeneity between ontologies
Local sources: DL ontologies
Alignments: *mappings*, view point of the target ontology
Semantics: CADS variant 1
Reasoning: distributed in peer-to-peer system
Drago distributed reasoner [30]

Formalism: $\mathcal{E}$-connection
Motivation: ontologies combination
Local sources: Logic ontologies with Abstract Description System
Alignments: *links*, view point of the target ontology
Semantics: CADS variant 2
Reasoning: distributed
Extended Pellet reasoner [31]

Formalism: P-DL
Motivation: ontologies import
Local sources: DL ontologies
Alignments: *foreign term*, view point of the target ontology
(*e.g.*, $\mathtt{O_i} \overset{\mathbf{t}}{\to} \mathtt{O_j}$) ontology O_j imports term t defined in ontology O_i
Semantics: CADS variant 1
Reasoning: distributed
P-DL distributed reasoner: `https://sourceforge.net/projects/p-dl-reasoner/`

Formalism: IDDL
Motivation: resolution of semantic heterogeneity between ontologies, mediation of alignments
Local sources: DL ontologies
Alignments: *mappings*, global view point
Semantics: CAIS
Reasoning: distributed
Draon distributed reasoner [12]

Formalism: $E-\mathcal{SHIQ}$
Motivation: resolution of semantic heterogeneity between ontologies, ontologies combination
Local sources: DL ontologies
Alignments: *mappings*, *links*, view point of the target ontology
Semantics: CADS combination of variant 1 and variant 2
Reasoning: distributed
$E-\mathcal{SHIQ}$ distributed reasoner [32]

Formalism: MLNK
Motivation: resolution of semantic heterogeneity between ontologies and alignments, ontologies combination,
Local sources: nodes hierarchically composed of aligned sub-nodes, independent of any language
Alignments: *mappings*,*links*, ontologies-pair view point
Semantics: DL-approach: ENCACS
: DDL-approach: DACOSLS
Reasoning: centralized for DL-approach
: distributed for DDL-approach
Reasoner: DLMLNKR [25], DDLMLNKR

Table 6. Summary table of state of art