VRIJE UNIVERSITEIT BRUSSEL

FACULTEIT WETENSCHAPPEN

VAKGROEP INFORMATICA EN TOEGEPASTE INFORMATICA

SYSTEMS TECHNOLOGY AND APPLICATIONS RESEARCH LAB

# STAR Lab Technical Report

# Creating a “DOGMAtic” multilingual ontology infrastructure to support a semantic portal

Jan De Bo, Peter Spyns & Robert Meersman



| | |
|---|---|
| affiliation: | |
| keywords | ontology representation and storage, multilinguality, context |
| number | STAR-2003-13 |
| date | 3/09/2003 |
| corresponding author | Peter Spyns |
| status | final |
| reference | R. Meersman & Z. Tari et al. (eds.), On the Move to Meaningful Internet Systems 2003: OTM 2003 Workshops, LNCS 2889, Springer Verlag, pp. 253 - 266 |



Pleinlaan 2, Gebouw G-10, B-1050 Brussel
Phone: +32-2-629.1237; Fax: + 32-2-629.3819
http://www.starlab.vub.ac.be

# Creating a "DOGMAtic" multilingual ontology infrastructure to support a semantic portal

Jan De Bo, Peter Spyns, and Robert Meersman

Vrije Universiteit Brussel - STAR Lab
Pleinlaan 2, Gebouw G-10, B-1050 Brussels, Belgium
{Peter.Spyns,jdebo,meersman}@vub.ac.be
http://www.starlab.vub.ac.be

**Abstract.** We present extensions to the current DOGMA ontology engineering framework so that the DOGMA ontology server is better equiped to cope with context and multilinguality, issues that are important when deploying "ontology technology" in a setting involving human users. An obvious example is a semantic portal that offers search facilities to its users. A short presentation of the DOGMA ontology engineering approach will be provided, before the extensions mentioned will be presented.
*Keywords: ontology and database modelling, context, multilinguality*

## 1 Introduction

More and more semantic portals, OntoWeb [33], KAON [23], OLR [7], or semantic based query methods, e.g. [5], are becoming popular on the internet or rather the semantic web. Our claim is that ontologies need to be linked with linguistic knowledge since the "eyeball web" with its human users is and will remain an important part of the semantic web (with its intelligent sofware agents). An important issue is the interaction with a user who wants to look up all kinds of information offered by a portal. A semantic portal provides for, amongst other things, the "visual input side" of a semantic retrieval engine [1] that is connected with an ontology server. Users can browse the underlying ontology to construct their query, which can become problematic with a large and complex ontology. The most easiest way for a human user to express a query is through natural language terms - as is the case for classical search engines.

The goal of this paper is to present some extensions on the current DOGMA (Developing Ontology-Guided Mediation for Agents) framework for ontology engineering [17] that are in with line the usage of natural language terms for semantic querying. We believe that the incorporation of "contexts" and "natural language terms" in an ontology is needed to correctly determine the meaning of a user query. The introduction of bridge types will allow to resolve issues about linking variables local to an application to ontology vocabulary.

[1] The presentation of the search results will not be considered here.

First, a general background of the field is given with a discussion of a specific semantic portal (see section 2). Since the underlying knowledge model of our ontology research is based on the DOGMA approach, we provide in section 3 an overview of the DOGMA initiative, a VUB STAR Lab research project on ontology engineering. Subsequently, we present a proposal for a more sophisticated treatment of contexts (section 4) and multilinguality (section 5). The introduction of bridge type definitions (see section 6) should allow to link information system applications to ontologies. Related work is discussed in section 7. Some indications on future work and final remarks (section 8) conclude this paper.

## 2 Background

Semantic portals today basically allow a user to define his/her query by means of structured input, i.e. a user selects terms (values) from drop down boxes that relate to properties of a concept in the underlying ontology. An important user interface problem is how to graphically represent a (large) ontology so that a user can easily browse and compose his/her semantic query (i.e. selecting the appropriate concept(s) and values). Several types of graphical representations of an ontology are studied, e.g. a tree [25], a hyperbolic view [34]:p.482] or a landscape view [30]. Sometimes a seemingly "regular" search box - e.g. the OntoWeb portal [25, 33] - "hides" a semantic search box if the search is done over the meta-data (ontology-based annotations) instead of the "raw" data. Notice that a user has to be familiar with the vocabulary of the ontology (i.e. consisting of ontology terms, opposed to intuitive natural language terms).

The metadata [2] is stored by back-end ontology servers - e.g. Sesame based on RDF [2]. An overview of ontology servers in general is given in [6]. None of these has a thoroughly developped view on multilinguality. This means that there is no, or only a very superficial, difference between a linguistic term that lexicalises a concept and a logical term that represents a concept in an ontology. Synonyms and translations become problematic in such an approach. We believe that multi-lingual language terms should be linked to a concept that is uniquely labelled by a specific natural language head term or an artificially constituted concept label. By doing so, the conceptualisation achieves a high degree of language independence. The back-end ontology server should thus be designed and equipped accordingly.

At first sight, the KAON Portal is an exception as it is said to cope with homonyms (and synonyms ?) [23]. However, it is not so clear to what extent its "Lexical Ontology Instance Model" (LOIM) is actually integrated in the KAON portal [3]. Also, the use of "context" to disambiguate terms (see section 3.3) is not considered, or at least not further explained. Finally, the KAON method to "lift" databases to the ontology level [35] seems to be rather focused on lifting a single database of which the conceptual model largely determines the ontology, rather then providing techniques to allow multiple databases (or other applications) to

[2] The raw data remains at its original location.
[3] The LOIM is not mentioned anymore in the remainder (examples) of the paper [23].

commit their internal model to a single domain ontology that is supposed to subsume individual applications or databases (see [32] for a related discussion).

## 3 DOGMA

Most often an ontology is defined as an *explicit, formal specification of a shared conceptualisation of a certain domain* [10], or as a *logical theory accounting for the intended meaning of a formal vocabulary* [13]. A DOGMA inspired ontology is based on the principle of a double articulation: an ontology is decomposed into an *ontology base*, which holds (multiple) intuitive conceptualisation(s) of a domain [4], and a layer of *ontological commitments*, where each commitment holds a set of domain rules to define *a partial semantic account of an intended conceptualisation* [12]. The DOGMA approach of VUB STAR Lab is based on three rather evident observations that:

1. agreements become easier if the items involved are simpler
2. most relevant human knowledge is massively available in natural language, in text documents and other "lexical" sources such as databases
3. conceptualisations - and hence ontologies - should be as independent as possible of intended application and design(er) context, and of the language used to reach the agreement

As these requirements however are mutually counteracting except for the most trivial of domains, a heuristic approach is adopted based on the extensive practical experience of database technology under the *model-theoretic* perspective for relational databases [28]. As a summary, one can state that the DOGMA approach takes agreed semantical knowledge out of an application by making use of an external ontology (and achieves a form of "meaning independence") [20]. This is done in much the same way that databases take data structures out of regular programs ("data independence") - see [17, 32] for details on DOGMA.

### 3.1 DOGMA ontology base

Currently, the ontology base consists of sets of intuitively plausible conceptualisations of a real world domain where each is a set of context-specific "representationless" binary facts types, called *lexons*, formally described as $< \gamma \quad term_1 \quad role \quad co-role \quad term_2 >$, where $\gamma$ denotes the context, used to group lexons that are logically related to each other in the conceptualisation of the domain [20]. Informally we say that a lexon is a fact that may hold for some application, expressing in that case that within the context $\gamma$ the $term_1$ (or head) may plausibly have $term_2$ (or tail) occur in an associating $role$ (with $co-role$ as its inverse) with it. Lexons are independent of specific applications and should cover relatively broad domains. Lexons in a DOGMA ontology base are always "true", i.e. free of further interpretation. E.g., *"bookstore: book is_identified_by/identifies ISBN"* is a lexon, with "bookstore"=$\gamma$, "book"= head, "ISBN"= tail, "is_identified"= role and "identifies" = co-role.

[4] Guarino would call this an *uninterpreted ontology.*

### 3.2 DOGMA commitment layer

The commitment layer, mediating between the ontology base and applications, is organised as a set of ontological commitments [11], each being an explicit instance of an (intensional) first-order interpretation of a task in terms of the ontology base. A commitment is a consistent set of rules (or axioms) in a given syntax that specify which lexons of the ontology base are visible (*partial account*) for usage in this commitment and that semantically constrain this view (i.e. the visible lexons). The rules that constrain the relations between the concepts (*semantic account*) of the ontology base are specific to an application (*intended conceptualisation*) using the ontology. Experience shows that agreement on the domain rules is much harder to reach than on the conceptualisation [21]. E.g., it is easy for people to agree on the binary fact represented by the lexon *"bookstore: book is_identified_by/identifies ISBN"* in a "bookstore" context, while they might disagree whether or not for a *given* application the ISBN number is a *mandatory* property of a book. An application (e.g., semantic portal) can adopt more than one commitment. A commitment, in principle, can be shared by several applications. Commitments can also include derivation and inference rules. Sets of ontological commitments can be regarded as reusable knowledge components (e.g., several applications can use the same "search commitment rules").

### 3.3 DOGMA contexts

Contexts have the generic property of disambiguating the lexical meaning of terms inside a lexon. Until now a context is represented by a symbol $\gamma_i \in \Gamma$, where $\Gamma$ is the context space of the domain to be modelled. Currently, $\gamma_i$ is a mere label that refers in a non formal way to a source (e.g., a document that contains and "explains" how the various terms are used in that particular context).

Two terms $t_1 \in T$ and $t_2 \in T$ are synonyms when they identify the same concept within the same context $\gamma$. When the meaning of identical terms is different, we speak about homonyms. For example, the term *bank* has a different meaning in a *geographical* context than in an *financial* context and therefore evokes different concepts. We refer to the classical semiotic triangle [24]: a word or term is a symbol in natural language evoking a language-independent concept in our mind that refers to a world object (referent). A term only indirectly (via the corresponding concept) "points" to a referent. The "evokes" relationship between a term and a concept has a cardinality of m:n.

## 4 Extensions to DOGMA

### 4.1 Extension of the context definition

We now redefine a context $\gamma_i \in \Gamma$ as a semantic cluster of concepts that are logically and meaningfully related. It is the task of the ontology engineer to build these contexts. To establish a relationship between terms and concepts in a given context $\gamma_i$, we define a *context mapping* $\psi_i$, from a domain $T$ (the set of terms) to a range $C$ (the set of concepts within that particular context $\gamma_i$), formally noted as $\psi_i : T \rightarrow C$, so that $range(\psi_i) = \gamma_i$. This is a m:1 mapping, because several terms can be associated with the same concept. A context mapping $\psi_i$ thus determines the meaning of terms by establishing a formal link between lexical terms and its underlying concept in a certain context $\gamma_i$. By definition the following equation holds $\Gamma = \bigcup_i \gamma_i = \bigcup_i range(\psi_i)$.

### 4.2 Representation of concepts

According to the DOGMA approach, terms are part of a *lexon* and are represented by natural language *words* in the ontology base. To describe a concept we propose to associate with each concept a *set of synonymous terms*. The idea of using a *synset* (=set of synonyms) to express the semantics of a concept is inspired on the approach taken in WordNet [8].

Wordnet offers two distinct functionalities: a *vocabulary* to disambiguate terms (=to describe the various senses of a term) and an *ontology* to describe the semantic relationships among the various senses. Wordnet can be seen as an early precursor of ontology development: it includes semantic relations among the synsets in the ontology of Wordnet as hyperonymy, hyponymy, meronymy next to antonymy and value_of (referring to the noun database of WordNet.)

An example will clarify: the natural language **term** *"java"* can "point to" different meanings according to the specific context it is used in. If we use it in the context of *"Computer Science"*, the term [5] *"java"* refers to a platform independent, object-oriented programming language. If it is used in the context of *"travelling"* it refers to an island in Indonesia and when used in the context of *"consumer goods"* it refers to coffee. The concepts lexicalised by the term *java* ($t =$ "$java$" $\in T$) will be denoted as follows in the respective contexts:

$$\gamma_1 = "CompSc"; \psi_1(t = java) = c \equiv \{"java", "programming\_language"\}$$
$$\gamma_2 = "ConsGoods"; \psi_2(t = java) = c' \equiv \{"java", "coffee", "beverage"\}$$
$$\gamma_3 = "Travel"; \psi_3(t = java) = c'' \equiv \{"java", "island", "vacation\_destination"\}$$

It is our aim to describe concepts by defining them as a set of semantically equivalent terms. The equivalence sign " $\equiv$ ", is used to describe the semantics of a concept.[6].

Formally we state that: $\psi_i(t) = c \equiv \{t, t', t'', t'''\}$, where $t, t', t'', t''' \in T$ and $c \in \gamma_i$.

This specification allows a machine to retrieve, compare etc. concepts. These unique combinations of synonymous terms describe the logical vocabulary we use

[5] One has to remember the difference between a natural language or lexical term and a logical term (or ontology term), which sometimes in the AI literature appears as a synonym of concept.

[6] It is thus obvious that the lexical term "java" is included in the set of synonymous lexical terms describing the related concept, which might also be labelled by the string "java". The latter is a logical term while the former is a natural language term - see the previous footnote

to model the given domain. Because ontology engineering often concerns rather specific domains (e.g. complaint management, European directives, surgery) to be modelled, we cannot only rely on Wordnet's vocabulary since it exclusively includes the 95.000 most common English words and lacks very specific or technical terms. Therefore we prefer to build our own vocabulary, separated but nevertheless compatible with and building on Wordnet.

Besides grouping semantically equivalent terms in order to uniquely and unambiguously describe the concepts occurring in the ontology, additional phrases (explanatory glosses) are needed to clarify a meaning or a concept. A gloss can contain a natural language description and/or a formal language definition (e.g. using Conceptual Graphs [31]). One can consider the gloss as the definition by intension. In practice, the natural language gloss will often be taken from one of the resources the domain expert has at his/her disposal when modelling the domain. In the realm of the previous example and considering the context of *"Consumer Goods"*, a gloss for the concept $c'$ represented by the term $t =$ *"java"* could be *"a beverage consisting of an infusion of ground coffee beans; he ordered a cup of coffee"* [38].

### 4.3 Representation of Context Space

In later work based on the pioneering research of McCarthy [19], Buvac concludes that common sense knowledge is needed to completely *lexically disambiguate* a proposition like *Vanja is at a bank* [4]. As a consequence, trying to resolve lexical ambiguity using a formal theory for context like McCarthy and Buvac propose does not seem to be the most practical way, at least for our purposes, to deal with the phenomenon of context. Indeed, it would require a human to explicitly add common sense knowledge to the reasoning system to solve a lexical ambiguity.

Since we have redefine a context ($\gamma_i \in \Gamma$) as the range (set of values) of one particular context mapping $\psi_i$, it can be formally interpreted as a particular (i.e. meaningfully related) set of concepts, which on their turn are labelled by a specialised domain vocabulary and explanatory glosses. Note that the idea of the "corpus based word relator" of Mitra and Wiederhold [22] is a similar attempt to disambiguate terms (in view of ontology integration), although they did not make the difference between a language term and an ontology term or concept (see section 6.3).

Our current thinking is to organise all these contexts ($\forall \gamma_i \in \Gamma$) in a tree structure. Each node in this tree thus represents one particular context $\gamma_i$. Because each separate context is a set of meaningfully grouped concepts, the context tree is populated with concepts. The tree is layered starting from the top with the $0^{th}$ *level*. This level contains the most common contexts of the context space. The $0^{th}$ level is then expanded by the contexts of the first level. Each context of the first level is derived from a context of the $0^{th}$ level, which means that the concepts of the first level context supplement and further refine (but not replace) the concepts of the $0^{th}$ level context. Let us assume that $\gamma_i^1$, a context from the first level, is derived from $\gamma_j^0$, a context from the $0^{th}$ level in the context tree. The set of concepts of $\gamma_i^1$ expands the set of concepts of $\gamma_j^0$. Formally we state that, $\gamma_j^0 \subset \gamma_i^1$. The same logic holds for all the other levels in the tree. We can thus generally state that if a context $\gamma_x^{n+1}$ of the $(n+1)^{th}$ level is derived from a context $\gamma_y^n$ of the $n^{th}$ level, then $\gamma_y^n \subset \gamma_x^{n+1}$.

Context trees may be helpful to describe the precise and unambiguous meaning of terms. This may be desirable for various reasons. For instance, a domain expert may want to determine the unambiguous meaning of a term, which he found in a resource describing a particular domain. It may be the case that a lot of words (i.e. terms) in the close environment of that particular term determine its context. This will often lead to a specific context node in the context tree since a context is a set of logically related concepts. If this context is not specific enough to determine the precise meaning of the term, the domain expert will have to descend one level in the tree (e.g. from level n to level n+1). If the meaning of the term is not found at all, the ontology engineer will have to expand the context tree at the node where the search activities stopped. In other cases (e.g. ontology integration) the tree will be more likely used by a software agent instead of a domain expert. We visualise the idea of a context tree by means of the example depicted in Figure 1.

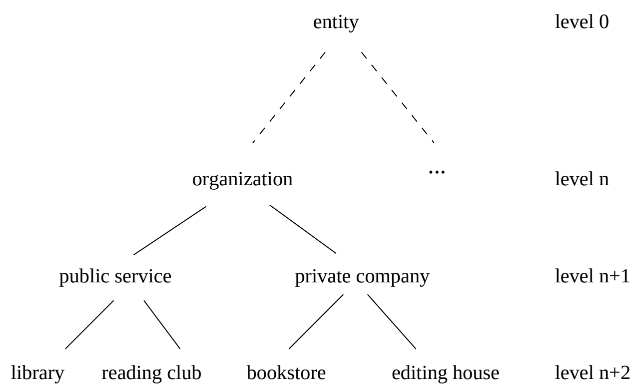


**Fig. 1.** Visualisation of a context tree

Note that the context tree is by no means complete. Real context trees would probably contain more subtrees. Also for spatial considerations, the figure does not contain any accompanying concepts for the contexts being represented. In figure 1 the *library* context extends the *public service* context with additional concepts.

It is our intuition that one could reuse (parts of) an upper level ontology to build a context tree. Contexts can be seen as a more general and coarser way of structuring a conceptualisation. As a consequence, a context might be equivalent to a certain extent with upper ontologies. However, in our approach, a context tree remains a separate entity (not merged or aligned with the ontology proper). For time being, we stick to a tree structure. Future work will provide an indication on the necessity of transforming the tree into a lattice.

## 5 Multilinguality

The DOGMA meta-model does not yet foresee in a multilingual representation of concepts. Until now we assumed the modelling language was English and used a term $t$ to *label* a concept $c$. However, when mining resources in order to build an ontology one cannot expect that all these resources occur in one and the same language. A classical, non technical, example is the following: in the context of "smoking", the French term *"feu"*, the English term *"light"* and the Dutch term *"vuur"* identify the same concept. In the context of a house, the English term "light is translated in French as "lumière" and "licht" in Dutch. The Dutch term "vuur" corresponds to the French "feu" that is translated in English by "fire". As one can see in Table 1, not all notions are lexicalised in a language by a different term, and the "meaning distribution" as a result of the lexicalisation process is not the same for all languages. Other similar examples can be found in [29]:p.84]. Therefore we want to introduce a new *linguistic identifier*, called $\lambda \in \Lambda$, where $\Lambda$ is the linguistic space.

| | English | French | Dutch | |
|---|---|---|---|---|
| $\gamma_1$ | *fire* | *feu* | *vuur* | $\rightarrow c_1$ |
| $\gamma_2$ | *light* | *feu* | *vuur* | $\rightarrow c_2$ |
| $\gamma_3$ | *light* | *lumière* | *licht* | $\rightarrow c_3$ |

**Table 1.** Example of a different meaning distribution for translated terms

Now we can extend our definition of a *concept* as introduced above in section 3.3. In a given context $\gamma \in \Gamma$, any concept can be represented by multiple terms in multiple natural languages. Each natural language corresponds with an identifier $\lambda_i \in \Lambda$. In our example, $\lambda_1$ corresponds to Dutch, $\lambda_2$ corresponds to English and $\lambda_3$ corresponds to French.

With a given context $\gamma_j \in \Gamma$, which is equal to "smoking" in our example, we associate the context mapping $\psi_j$. We now define: $\psi_j \lambda_i t_i \rightarrow c \quad with \lambda_i \in \Lambda$. In our example holds, $t_1$="vuur", $t_2$="light" and $t_3$="feu". These terms are lexical representations in different languages of the concept $c$ that we will denote as follows in our vocabulary: $c$={{"lighter", "light", "igniter", "ignitor"}; a device for lighting or igniting fuel or charges or fire; "Do you have a light?"}. We hope to apply the same mechanism to conceptual relationships.

¿From the above, it follows that lexons are a necessary but intermediary (since language dependent) representation. It echoes Guarino's statements that currently in AI the term "ontology" collapses the language independent conceptualisation level with the language dependent ontology level [13]:p.8]. Of course, we have to understand the term "ontology" in the way Guarino has circumscribed it - see [12]. A DOGMA ontology base (or rather "a conceptualisation base") eventually should consist of binary "conceptons" [7] formalised as $(concept_1 \quad relation \quad co-relation \quad concept_2)$ instead of lexons [8]. As a corollary, the commitment layer puts constraints on the conceptons and thus gains in scope. However, more research on this topic within the DOGMA framework is needed (e.g., how does multilinguality affect a context tree: is it enough to include the terms of other languages in the set of terms that define a concept?).

## 6 Bridging the semantic gap between an information system and a DOGMA ontology

### 6.1 NIAM bridge types

In order to restrain the ontology base (basically selecting lexons), applications (information systems) select sets of particular commitments, stored in the commitment layer. To express these links, we use *bridge types.* This terminology is taken from an information systems modelling methodology called aN Information Analysis Method (NIAM [37]), that evolved later to Object Role Modelling (ORM [14]).

In NIAM, bridge types are defined as binary fact types between lexical objects and non-lexical objects. We adopt the NIAM philosophy of a strict separation between lexical objects (LOTs) and non-lexical objects (NOLOTs). A lexical object or label is *an object in a certain reality which can be uttered, written down, or otherwise represented* [o.c.]. LOTs always consist of letters, numbers, symbols or other characters. They can be used as names for or references to other objects. A non-lexical object (NOLOT) is *an object in a certain reality which cannot be uttered, written down or otherwise represented. Non-lexical objects must be named by lexical objects or referred to by means of lexical objects.* [o.c.].

### 6.2 Bridge types from the Information Systems to the Ontology level

In practice, establishing the bridge between LOTs and NOLOTs explicitly will often boil down to a manual exercise. As an example, there could exist a bridge type between "isbn_no" (LOT) and "isbn" (NOLOT), "pub_date" (LOT) and "publishing_date" (NOLOT), "salary" (LOT) and "salary_amount" (NOLOT), etc. We also note that because the fields of database tables are mostly arbitrarily named, it will be very hard to automate the activity of establishing bridge types.

[7] Other denominations could be meta-lexon or conceptual lexon.

[8] We temporarily discard the issue of how to label a concept, e.g. numerical identifier vs. specific compositional artificial language expression vs. preferred natural language terms.

### 6.3 Bridge types from the Ontology to the Conceptualisation level

For the same reasons as we have introduced bridging types on the information system level, we will now apply this *bridging-logic* on the ontology-level (term-concept relation that we have introduced in section 3.3). We have defined the combination of a context mapping ($\psi_i$) and languages identifiers ($\lambda_i \in \Lambda$) as a mapping between terms (LOTs) and concepts (NOLOTs). Therefore this mapping forms a bridge type between lexicals and non-lexicals on the ontology level.

A very important note has to be made regarding the dual interpretation of a term in the ontology base. In the context of an information system, terms fulfill the role of NOLOTs whereas in the context of ontologies terms fulfill the role of LOTs because they are the lexical representations of the concepts (NOLOTs) in the ontology base. We visualised this distinction by means of Figure 2 that integrates through a flattened semantic triangle an information system and an ontology. The real world objects (referents) that are included in the information system can only be referenced by linguistic symbols (e.g., a term as a column name). However, as the semantics of these terms is implicit and therefore impossible to be shared amongst computers, the exact and shareable semantic definition (or intended meaning) of these terms (which can be different for each application - consider PN, PersName, PersonName, Name, Naam, Nom, ...) is defined on the logical level. Bridging between terms and concepts is done in the way explained in section 3.3. The same rationale is applied in case of terms in another natural language (see section 5).

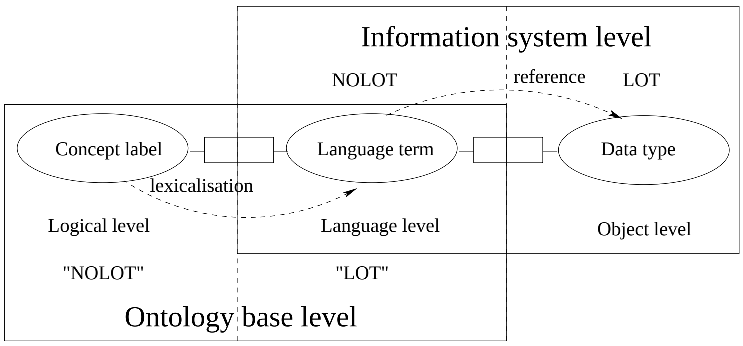


**Fig. 2.** Schema Overview

Perfect 1:1 matches (via bridges to the same concept) between terms of different languages and/or applications will not always be possible (e.g., see Table 1). Therefore, it is expected that specific equivalence or mapping rules will have to defined that are typical of a particular application (e.g., "pub_date" → "publishing date"), and as such, not re-usable by similar applications. These rules are located in a separate and new layer: *the application layer*. The commitments, as they are defined on the logical level, are independent of the local vocabularies

- implying that LOTs are not allowed in a commitment - and thus in principle re-usable by similar applications [9]. This is depicted in Figure 3.

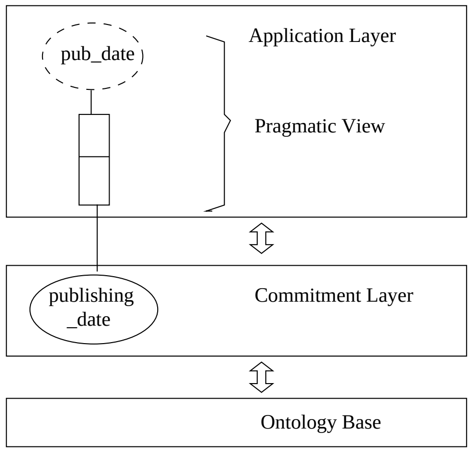


**Fig. 3.** existing DOGMA architecture extended with an Application layer

## 7 Related Work

In addition to the related work already mentioned in section 2, we mention in this section other efforts on particular aspects of related to the work presented here. In [9], it is shown how contextual reasoning can be formalised as deduction in a system allowing multiple first order theories. Within the BUSTER project [3], a context theory is defined as a collection of linguistic expressions, providing an explicit description of the domain. Another approach on contexts was taken by Lenat [18] who distinguished 12 categories in a context space. He incorporated this idea in Cyc, the world's largest knowledge base at this moment.

Pazienza and Vindigni [26] suggest to use natural language terms related to concepts to achieve an agreement on a shared representation of a domain. The context in which the terms are used helps to select the correct concept. One of the intended uses of the DOGMA ontology server is to function as a mediator. As such, the topics described in this paper blend in with the research on research on data mediators in general - e.g. see [1, 36]. Mediator systems are presented in [27].

[9] Although it could be possible to use LOTs inside of a commitment (implying the inclusion of the application layer in the commitment layer), a loss of reusability perspectives of a commitment is the net result since a commitment is now linked to the local vocabulary of a specific application.

Bridge types are well known in the ORM data modelling field, but have not yet been introduced in the field of ontology engineering before, at least to our knowledge.

Computational lexicographers are now also turning to the semantic web, as is illustrated by the creation of the MILE lexical model by the ISLE group, e.g., see [16]). Lexical semantics constitutes the natural complementary research area to be combined with research on ontologies in view of linking local terms of a conceptualisation to natural language terms .

## 8 Future work and Conclusions

In this paper we have proposed some extensions to the existing DOGMA framework. We also stressed the multilingual aspect of ontologies. The next step is to redefine the meta-model of the DOGMA ontology server and implement the additions and refinements mentioned. Practically speaking, this will result in a powerful ontology server that combines conceptual with multilingual lexicographic/terminologic information. It is our intention to replace the parts of the OntoWeb semantic portal for which VUB STAR Lab is partly responsible [25] by this new ontology/lexicon server combination to allow for genuine semantic underpinned multilingual term based searching. Of course, once language terms are linked to ontology concepts, a plethora of natural language processing techniques and modules (e.g., robust parsers, dialogue modules, multimodal input, ...) can be combined to create genuinely intelligent information providing agents and applications offering an intuitive and user friendly way of interaction.

**Acknowledgments**: A major part of this work has been funded by the IWT (Institute for the Promotion of Innovation by Science and Technology in Flanders): Jan De Bo has received an IWT PhD grant (IWT SB 2002 #21304) while Peter Spyns is supported in the context of the OntoBasis project (GBOU 2001 #10069). We would also like to thank our colleagues at STAR Lab for their stimulating discussions during the "Dagstuhl sessions", as well as the reviewers.

## References

1. Bergamaschi S., Castano S., Vincini M. & Beneventano D., (2001), Semantic integration of heterogeneous information sources, in *Data & Knowlegde Engineering* 36 (3): 215-249
2. Broekstra J., Kampman A., & van Harmelen F., (2002), Sesame: An Architecture for Storing and Querying RDF and RDF Schema, In Proc. of the First Internat. Semantic Web Conference (ISWC 2002), Springer LNCS 2342, pp. 54-68
3. Bremen University Semantic Translator for Enhanced Retrieval. *http://www.semantic-translation.de*
4. Buvac S. (1996). Resolving Lexical Ambiguity using a Formal Theory of Context. in Van Deemter and Peters (eds.), *Semantic Ambiguity and Underspecification*, CSLI Publications, Stanford

5. Chiang R., Eng Huang Chua C. & Storey V., (2001), A smart web query method for semantic retrieval of web data, *Data & Knowledge Engineering* 38: 63 - 84
6. Ding Y., Fensel D., Klein M. & Omelayenko B., (2002), The Semantic Web: Yet Another Hip ?, *Data and Knowledge Engineering* 41 (3): 205 - 227
7. Dhraief H., Nejdl W. & Wolpers M., Open Learning Repositories and Metadata Modeling, in *Proc. of the first Semantic Web Working Symposium*, 495 - 514
8. Fellbaum, C. (1998). Wordnet: An Electronic Lexical Database. Cambridge, US: The MIT Press.
9. Giunchiglia, F. (1993). Contextual Reasoning. *Published in Epistemologia : rivista italiana di filosofia della scienza*, 1993, vol: 16
10. Gruber, T.R.(1993). A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2).
11. Guarino N., Carrara M., & Giaretta P., (1994), Formalizing Ontological Commitments, in Proceedings of the 12th National Conference on Artificial Intelligence (AAAI94), AAAI Press pp. 560-568
12. Guarino N. and Giaretta P., (1995) Ontologies and Knowledge Bases: Towards a terminological clarification, in N. Mars (ed.), *Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing*, IOS Press A'dam, pp. 25-32
13. Guarino N., (1998), Formal Ontology and Information Systems, in N. Guarino (ed.), *Formal Ontology in Information Systems (Proceedings of FOIS 98)*, IOS Press, Amsterdam, pp. 3-15
14. Concept Schema and Relational Database Design, Prentice Hall Austria, 1995
15. Heflin J. & Hendler J., (2000), Workshop on Artificial Intelligence for Web Search, Papers from the AAAI Workshop. WS-00-01, pages 35–40. AAAI Press
16. Ide N., Lenci A., & Calzolari N., (2003), RDF Instantiation of ISLE/MILE Lexical Entries, in *Proceedings of the ACL 2003 Workshop on Linguistic Annotation: Getting the Model Right*
17. Jarrar, M., Meersman, R. (2002). Formal Ontology Engineering in the DOGMA Approach. in Meersman R., Tari Z. et al. (eds) *On the Move of to Meaningful Internet Systems 2002: CoopIS, DOA, and ODBASE; Confederated International Conferences CoopIS, DOA, and ODBASE 2002 Proceedings*, LNCS 2519, Springer Verlag, pp. 1238-1254,
18. Lenat D. (1998). The dimensions of context-space, Austin, TX: Cycorp Report
19. McCarthy J. (1993). Notes on formalizing context. *In Proceedings of the Thirtheenth International Joint Conference on Artificial Intelligence*
20. Meersman, R. (2001). Reusing certain database design principles, methods and techniques for ontology theory, construction and methodology, *STAR Lab Technical Report, http://www.starlab.vub.ac.be/publications/STAR-2001-01.pdf*
21. Meersman R., (2002), Semantic Web and Ontologies: Playtime or Business at the Last Frontier in Computing ?, in *NSF-EU Workshop on Database and Information Systems Research for Semantic Web and Enterprises*, pp.61 - 67.
22. Mitra P. & Wiederhold G., (2002), Resolving Terminological Heterogeneity in Ontologies, in *Proceedings of the Workshop on Ontologies and Semantic Interoperability at the 15th European Conference on Artificial Intelligence (ECAI 2002)*
23. Motik B., Maedche A. & Volz R., (2002), A Conceptual Modeling Approach for Semantics-driven Enterprise Applications, in Meersman, R. & Tari, Z., (eds.), in On the Move to Meaningful Internet Systems 2002: CoopIS, DOA, and ODBASE; Confederated International Conferences CoopIS, DOA, and ODBASE 2002 Proceedings, of series LNCS 2519, Springer Verlag
24. Ogden & Richards (1923), *The Meaning of Meaning.* 8th Ed. New York, Harcourt, Brace & World, Inc

25. Oberle D., & Spyns, P., (2003), The Knowledge Portal "OntoWeb", in Staab, S. & Studer, R. (eds.), Handbook on Ontologies in Information Systems, LNCS, pp. 521-540, Springer Verlag [in print]
26. Pazienza M.-T. & Vindigni M., (2002), Language-based agent communication, in *Proceedings of the Internation Conference Knowledge Engineering and Knowledge Management Workshop on Ontologies for Multi-Agent Systems* [http://www.csc.liv.ac.uk/ valli/Papers.html]
27. Rahm E. & Bernstein Ph., (2001), A survey of approaches to automatic schema matching, *The VLDB Journal* 10: 334 - 350
28. Reiter R., (1988), Towards a Logical Reconstruction of Relational Database Theory, in Mylopoulos J. & Brodie M.L., *Readings in AI and Databases*, Morgan Kaufman.
29. Resnik Ph. & Yarowsky D., (1997) A Perspective on Word Sense Disambiguation Methods and their Evaluation, in M. Light, *Tagging Text with with Lexical Semantics, why, what and how ? (Proc. of the ACL - SIG on the Lexicon Workshop)*, pp. 79-86
30. Sabol V., Kienreich W., Granitzer M., Becker J., Tochtermann K., & Andrews K., (2002), Applications of a Lightweight, Web-based Retrieval, Clustering, and Visualisation Framework. in Karagiannis D. & Reimer U. (eds.), *Proceedings of the Fourth International Conference on Practical Aspects of Knowledge Management (PAKM02)*, pp. 359–369, LNAI 2569, Springer Verlag
31. Sowa, J.F., (2000), *Knowledge Representation: Logical, Philosophical, and Computational Foundations*, Brooks Cole Publishing Co.
32. Spyns, P., Meersman, R., & Jarrar M. (2002). Data Modelling versus Ontology Engineering. in Sheth A. and Meersman R. (eds), *SIGMOD Record Special Issue on Semantic Web, Database Management and Information Systems* 31(4):12-17
33. Spyns, P., Oberle, D., Volz, R., Zheng, J., Jarrar, M., Sure, Y., Studer, R. & Meersman, R., (2002), OntoWeb - a Semantic Web Community Portal. Karagiannis, D. & Reimer, U., (eds.), in Proceedings of the Fourth International Conference on Practical Aspects of Knowledge Management (PAKM02), LNAI 2569, page 189-200. Springer Verlag.
34. Staab S., Angele J., Decker S., Erdmann M., Hotho A., Maedche A., Schnurr H.-P., Studer R., & Sure Y., (2000), Semantic community web portals, in Proceedings of the 9th International World Wide Web Conference, pp. 473-491, Elsevier
35. Stojanovic L., Stojanovic N. & Volz R., (2002), A reverse engineering approach for migrating data-intensive web site to the Semantic Web, in *Proceedings of the Conference on Intelligent Information Processing, World Computer Congress*, Kluwer Academic Publishers.
36. Wiederhold G., (1994), Interoperation, Mediation, and Ontologies, in *Proceedings International Symposium on Fifth Generation Computer Systems (FGCS94), Workshop on Heterogeneous Cooperative Knowledge Bases*, vol. W3, pp. 33 - 48, ICOT, Tokyo, Japan
37. Wintraecken J.J.V.R. (1990). The NIAM Information Analysis Method, Theory and Practice. Kluwer Academic Publishers, The Netherlands.
38. http://www.cogsci.princeton.edu/wn/ [Wordline 1.7.1 on-line search]