

Functional Wheels and Conceptual Brakes: Will your ontology take-off?

Bernard Vatant¹ et Florence Amardeilh^{1,2}

¹MONDECA, 3 cité Nollez, 75018 Paris, France
bernard.vatant@mondeca.com
<http://www.mondeca.com>

²MoDyCo (Modèles, Dynamique, Corpus), Université Paris X Nanterre, 92 Nanterre, France
florence.amardeilh@mondeca.com
<http://www.modyco.fr>

Abstract

Ontology management tools are now mature as a core technology. They have been proven to be flexible, scalable, so we know ontology-driven information systems can fly. But migration of legacy systems needs methodology and guidelines regarding the way legacy concepts will be represented and used in the target system in order. We propose here a functional approach, where the choice of representation framework is driven by the intended usage of concepts, and how focusing on a single representation can be a conceptual brake. We show how the terminological aspects can be integrated with the ontological ones, that different representations of the same concept have generally to coexist in the system for different usages, and how the various Semantic Web languages can be used to glue together those different representations. In particular we focus on the central role of thesaurus-like representations as providing the needed bridge between the various terminological and formal aspects of the same concept. At last, we show a few examples of ontology-driven systems that have actually take-off in the real life, based on this methodology.

1. Introduction

Building a so-called *semantic* or *ontology-driven* information system is not a process starting from scratch. Most of the concepts that will be made explicit and used in the target system are already implicitly or explicitly present in the legacy, in the form of terminology, controlled vocabularies, indexing categories, taxonomies, classification systems etc. The way those concepts will be migrated is critical, since the target system has to take into account their role in the legacy system in order to formalize them correctly. Too often, this migration is too much constrained by some a priori ideas about what concepts should look like in the target ontology, like "Every concept is a class", and "The hierarchy of concepts is a subsumption hierarchy".

2. Components of a semantic architecture

Bourigault & al. defined the notion of Terminological and Ontological Resources (TOR) at the crossroads of the Terminologies and the Artificial Intelligence areas, and more particularly in the Knowledge Engineering field [Bourigault, 2004]. This notion gathers different types of resources, from index and glossaries to ontologies, including also lexical databases and thesaurus. In that section, we present the three main TORs allowing the representation and the modeling of a domain knowledge: the terminologies, the thesaurus and

the ontologies¹.

2.1. Controlled vocabulary (terminology)

One way to describe a domain is to list the vocabulary that represents that domain. There are several ways to do so: lexicons, controlled vocabularies, reference table, glossaries, etc. They all represent different views of the same notion: the **terminology**. These terminological resources are very helpful to classify, search, translate or write documents for the concerned domain. They are also used in Natural Language Processing methods and tools to identify and extract the pertinent information of the domain. But they are not sufficient by themselves to formally represent the knowledge of the domain.

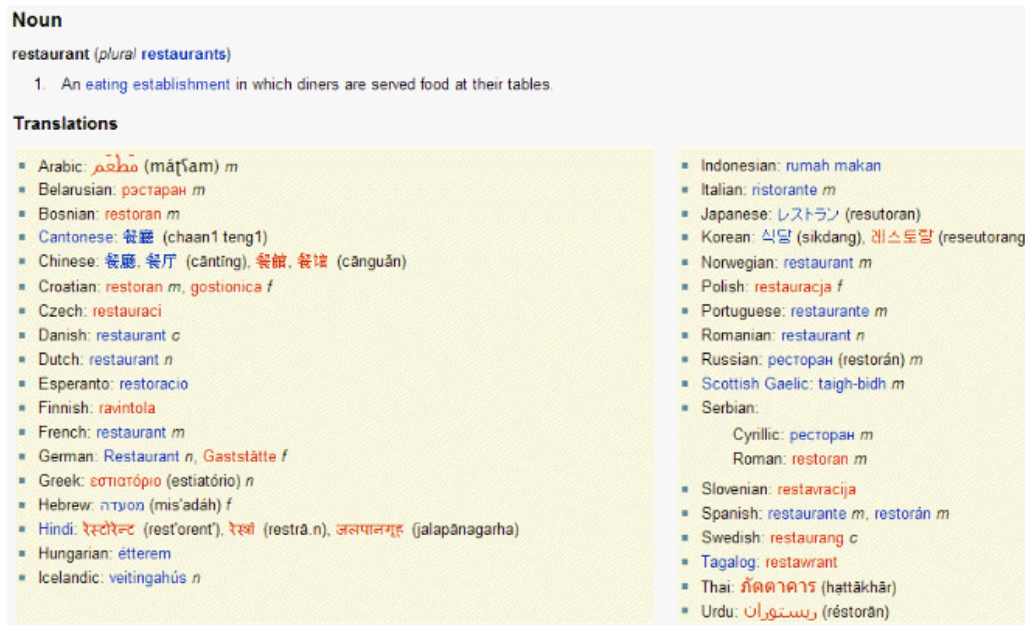


Fig.1: "restaurant" entry in the english Wiktionary

2.2. Structured vocabulary (thesaurus)

Since many years, terminologists and librarians have developed models to structure and organise the terminological resources. These models can be simple classifications, structured taxonomies, or more sophisticated thesaurus [Charlet, 2004]. Science always has the objective to identify and classify the objects of the world for studying and understanding them as done in Natural Science to describe the fauna and flora [Charlet, 2002]. The classification is usually set through a **taxonomy** that organises hierarchically the objects or the terms of the domain. This hierarchy corresponds to the relation of hyperonymy, structuring the terms of the vocabulary in which a term X has a broader sense (BT) or a narrower sense (NT) than a term Y. For example, in Fig. 2, the term "Restaurants" has a broader term "Catering" and two narrower terms "Gastronomic restaurants" and "Restaurant chains".

Besides, a **thesaurus** is defined as "a documentary language based on a hierarchical structure", knowing that a documentary language is "an organised set of normalised terms, used to represent the content of documents to be memorised for later search" [Bourigault, 2004]. Thus a thesaurus is considered as a controlled and structured vocabulary in which the relations between the terms of the domain are clearly specified, constituting a terminological network. In addition to the taxonomic structure of the thesaurus, constituting its backbone, a thesaurus models other types of relations such as synonymy, homonymy, and associative between the terms. Some attributes, such as a definition or an abbreviation, can also be added to the terms. As mentioned in its definition, the objective of a thesaurus is to facilitate document search by producing a

¹ See also the thesis of Audrey Baneyx [Baneyx, 2007] that presents the different existing TORs

consistent indexation² of the documents that points to the terms modelled, also called descriptors in that context.

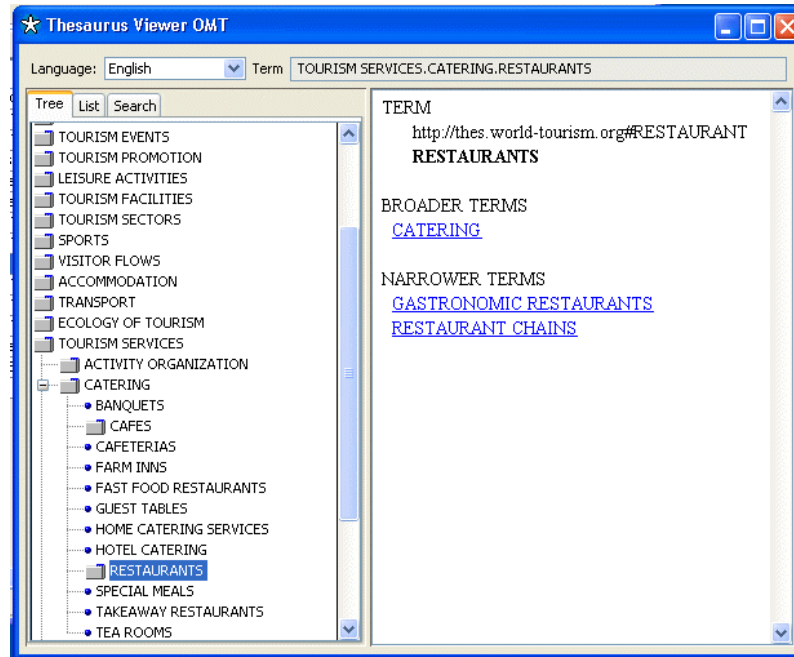


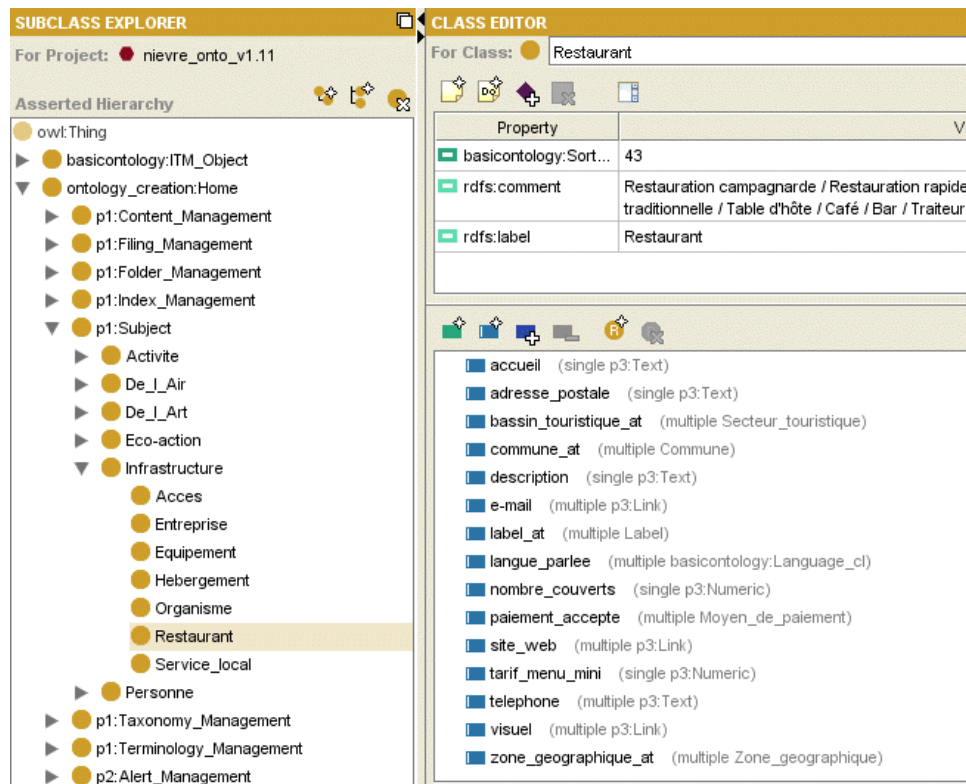
Fig.2 : "RESTAURANTS" descriptor in ThManager interface
Thesaurus of Tourism and Leisure Activities - World Tourism Organisation

2.3. Formal **vocabulary** (ontology)

Thanks to ontologies, we are reaching another level of domain representation: its concepts rather than its vocabulary. At the beginning of the 90s, researchers in Artificial Intelligence started to use that notion, originated from the field of Metaphysics, to formalise the knowledge. For them, what “exists” can be “represented”. In that context, they defined an **ontology** as an artefact allowing the representation of the existing by using a formal and consensual vocabulary. One of the first definitions of an ontology in Artificial Intelligence is the one formulated by Gruber [Gruber, 1993], reworked by Studer & al [Studer, 1998] as the “formal and explicit specification of a shared conceptualisation”.

More precisely, an ontology provide the means to identify the concepts of a domain by organising them hierarchically and by defining their semantic properties in a formal knowledge representation language. That language eases the sharing of a consensual view on that domain between the software applications that are using it [Bourigault, 2004]. Thus, an ontology is a structure consisting of a set of concepts (or classes), relations and attributes. The concepts, organised in a taxonomy, represent the objects, abstracts or concretes, elementary or composites, of the real world, such as the concept “Restaurant” in Fig. 3. The relations model the interactions between concepts whereas the attributes correspond to characteristics or specificities of a concept used to uniquely define that concept in the domain [Charlet, 2004]. The instances of concepts, also called individuals, are not part of the ontology, but rather of the knowledge base [Handschuh, 2005]. Actually, the knowledge base stores not only the instances of concepts, but also the instances of relations and the data values of the attributes accordingly to the ontology’s constraints.

² Cf. Norme ANSI/NISO Z39.19-1993 (R1998), p. 1.



*Fig.3 : "Restaurant" class in Protégé interface
Mondeca "Nièvre" ontology*

Until now, we described a first level of ontologies called the Conceptual Model. A second level allows the definition of more complex ontologies by making reference to Logic theories, and particularly those from Description Logics [Laublet, 2007]. They formalise a set of axioms and inference rules used to restrict the ontology model, to verify its validity and its consistency and to perform some reasoning actions on the domain. Most of the researches in Knowledge Engineering are looking forward to building such ontologies in order to express the semantic of the domain the more formal as possible. As such, in a concrete application, the goal of an ontology is clearly different from the one of a thesaurus: it is used to semantically formalise the way the domain operates, to infer new knowledge, to constraint the user interfaces or the reasoning mechanisms... Therefore it must also take into consideration the functional needs and objectives of the target application that will make use of it. Bachimont in [Chaumier, 2007] specifies that "ontologies are artefacts built as a function of a definite task and cannot be reused, as formal objects, for another task". On the other side, a thesaurus is more independent from the application as it just describes and organise the vocabulary of the domain. It can be more easily shared and reused by various applications based on the same domain; it is to say in different indexation contexts.

2.3. Complementarity of the Terminological and Ontological Resources

The Terminological and Ontological Resources do not offer the same level of knowledge representation: the terminologies are less expressive than the thesaurus, themselves less formal than an ontology. We would like to emphasize here the fact that thesaurus are not ontologies: they model the vocabulary of a domain but they do not provide a representation of the knowledge of the domain. On the one hand, the capacity of the thesaurus to express a controlled vocabulary as well as a set of lexical relations between the terms of that vocabulary constitutes a first level of semantic annotation. In that case the annotations are simple pointers to the thesaurus' descriptors. They are used as index entries to improve the results of the search engines that use this kind of annotations. On the other hand, the domain ontologies are more and more used to annotate the resources. In fact, a same resource can be annotated by different domain ontologies, offering different points of view on the same content. Not only the semantic annotations created from an ontology improve the

information search as they can rely on inference and reasoning mechanisms, but they can also be combined to enrich the knowledge base of the domain.

Actually, the thesaurus and the ontologies offer content accesses from different angles: the one of the vocabulary for the thesaurus and the one of the conceptualisation for the ontology. Consequently, they play complementary roles in knowledge engineering for indexation and annotation applications [Hernandez, 2005] and as resources for assisting the creation of ontologies [Charlet, 2004]. As Bachimont said, “the ontologies are not without connection with the terminologies; one can find in the thesaurus the resources to bootstrap an ontology. But we must be very cautious that they only consist of ‘resources for’ and not of ‘embryons of’” [Chaumier, 2007].

3. Integration and use of Semantic Web languages

Most semantic architectures will need to integrate the three above aspects. Therefore, for each concept involved, the choice of the type of representation must be guided by the functional objectives. And for some concepts several different representations, corresponding to different functional objectives, are likely to coexist. In the “Restaurant” example, a Tourism Agency system may need any or all of the following, as will be shown in the first example (section 4.1 below)

1. Terminological resources for "Restaurant", in order to correctly use the word and its translations in multilingual publications.
2. A *descriptor* "Restaurants" and its broader and narrower terms to index publications, and for faceted search on a public website.
3. A *class* "Restaurant" to represent, manage and query specific instances in a back-office knowledge base, with specific properties and relations (number of tables, menu, prices, opening hours ...)

The various languages of the RDF family will be used to support those different representations.

- ✓ SKOS provides the natural representation framework for (2).
- ✓ SKOS extensions could be able to link (1) and (2), depending if terms are eventually handled by SKOS independently of concepts (see 4.1.3 below)
- ✓ RDFS or OWL are languages relevant to the expression of (3).

(3) and (2) can be linked using OWL vocabulary, by anonymous classes based on restrictions on the value of an attribute. Instances of a class can be inferred from the values of an indexing attribute, or the other way round indexing rules can be based on semantic properties defined in the ontology. (see 4.1.1 below).

4. Examples

4.1. Mondeca examples

In this section, we present three actual uses cases met in Mondeca applications. Those use cases come from various domains of application, showing both the variety of functional requirements that can be met, and the genericity of the approach presented in section 3.

4.1.1. Management and publication of tourism resources

The customer is a regional tourism agency (Nièvre en Bourgogne) which uses Mondeca ITM to manage and publish information about its territorial offer : accomodations, restaurants, products and crafts, heritage. Those objects are described and managed by business experts in the agency back-office using a fine-grained OWL ontology as presented in Fig.3 above. For the front-end website, each object (instance) is published in a simpler XHTML format, and accessed by end-users using a faceted navigation, based on a taxonomy expressed in SKOS. Specific business rules based on the ontology features map objects to taxonomy entries. What is in the back-office an instance of the "Restaurant" class ends up in the front-end as a published page indexed against the "Restaurant" category, but also on its geographical location, spoken languages, quality labels ... The indexing process is quite straightforward in this case, but it can be based on more complex rules. The faceted navigation is mixed up with full-text search in product description.



Fig.4 : Selecting a restaurant in Nièvre, using integrated faceted navigation and full-text search

4.1.2. Dealing with multiple classifications

Exemple Bureau Veritas : navire "à passagers" et "Ro-Ro". Indexation et classes anonymes (on Property ... hasValue).

4.1.3. Dealing with multilingual terminologies

Exemple Lafarge : distinction descripteurs et termes pour la gestion multilingue

4.2. TAO case study : the ATA model

Cas générique : un système de classification hiérarchique à sémantique multiple

5. Conclusion

Since 2001, we have built ontologies for dozens of customers and projects on the behalf of Mondeca. To do so, we developed a pragmatic approach and identified a certain number of bottlenecks. Since it is easier to identify what does not work instead of finding the killer methodology, let's start by listing a those things that *do not work* when transitioning real applications towards a domain-oriented ontology model.

1. **Building ontologies “from scratch”** out of “tacit knowledge” of domain experts, using ontology editors. Domain experts have generally no or poor expertise in knowledge representation tools and languages. The learning curve is thus too steep to expect them to get such expertise quickly in a project life-time. And supposing domain experts have the needed expertise, working this way would be costly and most of the time wasting their precious time. In most cases it is likely to miss the target, if the task is not supported by a previous audit of the legacy of data, documents and schemas, and a specification of requirements for the target system.
2. **Re-using complex domain ontologies** built for the domain in similar projects. The more complex an ontology is, the more tied it is to its original context of development and use and the less likely it is to fit another context. It's often as difficult and costly to trim such ontologies in order to keep only the relevant parts than to re-build those parts completely.
3. **Building domain ontologies with a “top-down” approach** as extensions of “foundational ontologies”. Foundational ontologies are often highly abstract and constraining. Besides, they are almost never adapted to business requirements. They bear strong constraints that are rarely part of the requirements for the system to build.
4. **Building ontologies without knowing the technical and functional requirements** of the system that will implement them: meta-model, components, architecture, interfaces, queries, automatic population, publication, etc.
5. **Building ontologies without considering the legacy data** which will be used to populate the knowledge base(s) of the target system. At some point the explicit semantics of the target system will have to match the implicit semantics embedded in the data structure of the source. Otherwise migration of such data will be impossible.

Generally speaking, *what does not work is to build ontologies without a complete specification of the target system they will be part of*. Having in mind just a static knowledge representation is not enough. Ontologies are developed to be parts of a target information system and consequently, they have to fit smoothly with all components of this system, even more so as they are backbone components, upon which all other components will rely.

Besides, we can notice that there is no generic, standardized and consensual methodology existing for transitioning existing applications to a SOA based on domain ontologies. But rather several methodologies for building ontologies have emerged among which we can cite the ones described by Uschold [Uschold, 1995], by Gruninger [Gruninger, 1995], by Fernandez & Gomez-Perez in Methontology [Fernandez, 1997] [Blasquez, 1998] and by York Sure & al. in On-To-Knowledge [Sure, 2003].

The lifecycle of these methodologies is strongly based on the software engineering one and we can identify the following common steps:

- The specification / assessment of the application need
- The conceptualisation, i.e. the knowledge acquisition
- The formalisation, i.e. the ontology coding
- The integration of existing ontologies, thanks to their alignment or merging through mappings
- The implementation of the ontology into the target application
- The ontology evaluation, documentation and maintenance

Various methods, techniques and tools have been proposed to assist the knowledge engineers in the different tasks of the methodology lifecycle, such as:

- Extracting ontologies from texts [Bourigault, 2004] [Aussenac, 2000];
- Structuring the hierarchies of concepts and relations [Guarino, 1992] [Guarino, 2000] [Bachimont, 2001] [Kassel, 2002];
- Merging and adapting existing ontologies by using tools such as Onions [Gangemi, 1999] and Prompt [Noy, 2003];
- Collaborative development [Domingue, 1998], [Tolksdorf, 2005].

Nevertheless, we consider that the actual tools to automate the creation of ontologies are still far away from fulfilling the requirements described earlier. Moreover, they do not permit to solve the identified bottlenecks. To our point of view, the best they can provide are a terminology of the domain, in some cases a thesaurus, a taxonomic backbone of concepts based on the terminology and high-level properties (attributes and/or relations) between those concepts. But we agree that some of the proposed tools, such as OntoGen [Fortuna, 2005] or Terminae [Szulman, 2002] can bootstrap the ontology creation and really assist the knowledge engineer in the identification of the domain's concepts. Then, she can rework the suggestions made by the tools to restructure them, to enrich them and to add the restrictions, rules and axioms necessary for reasoning.

To conclude, we are convinced that the process of building ontologies, and by the same way of transitioning legacy systems to ontologies, still relies on a close collaborative work with the domain experts. Indeed the legacy systems are composed of an important part of implicit knowledge that requests the help of the domain experts to be made explicit. From that perspective it seems difficult to fully automate the transitioning process. Yet the use of some utility tools to integrate several heterogeneous legacy data sources and to transform automatically their formats into the target one, being the target ontology model, can alleviate the transitioning burden. What would highly assist the knowledge engineer in her task is a tool able to design the transition rules between the different data sources into the ontology format.

→ *peut-être rajouter un mot en conclusion sur la manière de modéliser qui doit être pris en compte par rapport à ce que tu as présenté dans tes parties... ?*

Acknowledgments

Thanks to TAO

Thanks to Dassault

Références

AUSSENAC-GILLES N., BIEBOW B. & SZULMAN S. (2000). Revisiting ontology design: a method based on corpus analysis, in *Proceedings of the European Knowledge Acquisition Conference (EKAW'2000)*, Springer-Verlag LNCS 1937, pp.172-188.

BACHIMONT B. (2001). Modélisation linguistique et modélisation logique des ontologies: l'apport de l'ontologie formelle, in *Actes des journées francophones d'Ingénierie des Connaissances (IC'2001)*, Presse Universitaire de Grenoble.

BANEYX A. (2007) Construire une ontologie de la pneumonie : aspects théoriques, modèles et expérimentations, *Thèse de doctorat*, Université Paris 6, 2007, 216 p.

BLAZQUEZ M., FERNANDEZ M., GARCIA-PINAR J.M. & GOMEZ-PEREZ A. (1998). Building Ontologies at the Knowledge Level using the Ontology Design Environment. In *Proceedings of the 11th Knowledge Acquisition Workshop (KAW'98)*, Banff, Canada.

- BOURIGAULT D., AUSSENAC-GILLES N. & CHARLET J. (2004). Construction de ressources terminologiques ou ontologiques à partir de textes : un cadre unificateur pour trois études de cas, In PIERREL J.-M. ET SLODZIAN M., Eds., *Techniques Informatiques et Structuration de Terminologies*, Numéro Spécial de la Revue d'Intelligence Artificielle (RIA), Vol. 18(1), Hermès, Paris, pp.87-110.
- CHARLET J., BACHIMONT B. & TRONCY R. (2004). Ontologies pour le Web Sémantique, In CHARLET J., LAUBLET P. & REYNAUD C., Eds., *Le Web sémantique*, Hors série de la Revue Information - Interaction - Intelligence (I3), 4(1), Cépaduès, Toulouse, pp.69-100.
- DOMINGUE J. (1998). Tadzebao and WebOnto : Discussing, Browsing, and Editing Ontologies on the Web. In *Proceedings of the 11th Knowledge Acquisition Workshop (KAW'98)*, Banff, Canada.
- FERNANDEZ M., GOMEZ-PEREZ A. & JURISTO N. (1997). METHONTOLOGY : from ontological art towards ontological engineering, in *Proceedings of the Spring Symposium Series on Ontological Engineering (AAAI'97)*, AAAI Press, Stanford, USA.
- FORTUNA B., Mladenic D. & Grobelnik M. (2005). Semi-automatic construction of topic ontology, In *Proceedings of the ECML/PKDD Workshop on Knowledge Discovery for Ontologies*.
- GANGEMI A., PISANELLI D. M. & STEVE G. (1999). An Overview of the ONIONS project : Applying Ontologies to the Integration of Medical Terminologies, In *Data and Knowledge Engineering*, Vol. 31(2).
- GRUNINGER M. & FOX M. S. (1995). Methodology for the design and evaluation of ontologies, In *Proceedings of the Workshop on Basic Ontological Issues on Knowledge Sharing (IJCAI'95)*, Montréal.
- GUARINO N. (1992). Concepts, Attributes and Arbitrary Relations : Some linguistic and ontological criteria for structuring knowledge bases, In *Data and Knowledge Engineering*, Vol. 8(3).
- GUARINO N. & WELTY C. (2000). A Formal Ontology of Properties, In *Proceedings of European Knowledge Acquisition Conference (EKAW'2000)*, Springer-Verlag LNCS 1937, pp.97-112.
- HANDSCHUH S. (2005). Creating Ontology-based Metadata by Annotation for the Semantic Web, *Thèse de doctorat*, University of Karlsruhe, 225 p.
- HERNANDEZ N. (2005). Ontologies de domaine pour la modélisation du contexte en Recherche d'information, *Thèse de doctorat*, Université Paul Sabatier de Toulouse, 248 p.
- KASSEL G. (2002). OntoSpec : une méthode de spécification semi-informelle d'ontologies, In *Actes des journées francophones d'Ingénierie des Connaissances (IC'2002)*, Rouen, pp.75-87.
- LAUBLET P. (2007). Web Sémantique et Ontologies, In *Nouvelles technologies cognitives et concepts des sciences humaines et sociales*, Vol. 1, Humanités Numériques, Hermès, Paris, to be published in 2007.
- NOY N. F. & MUSEN M. A. (2003). The PROMPT suite : interactive tools for ontology merging and mapping, In *International Journal of Human-Computer-Studies*, Vol. 59(6).
- SURE Y., AKKERMANS H., BROEKSTRA J., DAVIES J., DING Y., DUKE A., ENGELS R., FENSEL D., HORROCKS I., IOSIF V., KAMPMAN A., KIRYAKOV A., KLEIN M., LAU T., OGNJANOV D., REIMER U., SIMOV K., STUDER R., VAN DER MEER J. & VAN HARMELEN F. (2003). On-To-Knowledge: Semantic Web enabled Knowledge Management, In *Web Intelligence*, Springer-Verlag, pp.277-300.
- SZULMAN S., BIEBOW B. & AUSSENAC-GILLES N. (2002). Structuration de Terminologies à l'aide d'outils d'analyse de textes avec TERMINAE, in *Traitement Automatique de la Langue (TAL)*, Numéro spécial « Structuration de Terminologie », Vol. 43(1), Hermès, pp.103-128.
- TOLKSDORF R., NIXON L. J. B., LIEBSCH F., MINH NGUYEN D., PASLARU BONTAS E. & NIXON L. J. B. (2005). Enabling real world Semantic Web applications through a coordination Middleware, In *Proceedings of the 2nd European Semantic Web Conference (ESWC 2005)*, Heraklion, Greece.
- USCHOLD M. & KING M. (1995). Towards a Methodology for Building Ontologies. In *Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing (IJCAI-95)*, Montréal, Canada.