Scalability and Knowledge Reusability in Ontology Modeling

Mustafa Jarrar   **and**   **Robert Meersman**

Abstract-- **The thesis of this paper is to present and discuss the scalability and reusability capabilities of DOGMA, an ontology modeling approach. Ontologies are repositories of domain knowledge and essential for knowledge management in organizations and for achieving interoperation among information systems. In the DOGMA ontology server architecture we implement ontologies as classical database resources separating the "fact base" from the constraints, rules, derivations etc. that commit an application to such a given ontology "base". This separation allows an increased degree of scalability and reusability for the activities of ontology building. These issues are key in the context of the so-called Semantic Web where very large numbers of partial ontologies will emerge.**

Index Terms— **ontology, scalability, knowledge reusability, ontologies, ontology base, modeling, methodology, commitment layer.**

1. INTRODUCTION

Scalability and knowledge reusability are two important and critical aspects of ontology modeling approaches. An *ontology* for the purpose of this paper is a *shared* and *agreed* computer-based resource of domain knowledge, which enables interoperation between information systems (IS) in the broadest sense. Examples of simple ontologies appear further in the paper; concretely we shall implement ontologies as databases of possible facts on the one hand, and *rules* (derivations, constraints) on the other hand that allow applications to *commit* to those facts. The very long expected life-cycle of an ontology implies the importance of handling in a *scalable* manner (1) the (very) large size of its content, (2) the complexity of construction and growth, and (3) the diversity of multi-domain ontological content. Also, intersections between domains, and the high costs of building such ontologies implies the importance of design capabilities for maximization of *knowledge reusability*, for example, when building a new "Car Rental" ontology, we may reuse the "Payment" context from an existing "Shopping" ontology. In short, we argue that scalability and knowledge reusability must provide the essential

Mustafa Jarrar is a VUB STARlab researcher, Vrije Universiteit Brussel, 10G730, Pleinlaan 2, Brussels B-1050, Belgium, (e-mail: mjarrar@vub.ac.be).

Robert Meersman is the VUB STARlab director, Vrije Universiteit Brussel, 10G730, Pleinlaan 2, Brussels B-1050, Belgium, (e-mail: meersman@vub.ac.be).

methodological principles that underlie the modeling of an ontology.

The thesis of this paper is to discuss and illustrate the scalability and reusability capabilities of the DOGMA framework [www.starlab.vub.ac.be/dogma], an ontology modeling and deploying approach, under development in VUB STARLab as part of a number of European and nationally funded projects. In this paper we will however not present the formalization of DOGMA, and only present its technical and architectural basis in passing as the details of these have been and will be explained in other and forthcoming papers.

STRUCTURE OF THIS PAPER: first of all section 2 will briefly give an overview of what ontologies are, and why they are needed in computer applications. Respectively, Section 3 and 4 will discuss in details issues related to the scalability and reusability of ontologies. Section 5 will overview the DOGMA approach and illustrate its capabilities of maximizing both scalability and reusability. Finally, section 6 will give a conclusion.

2. BACKGROUND

Computer science (re-)defines ontology[1] as a branch of knowledge engineering, where the agreed *semantics* of a certain domain is represented formally inside a computer resource, which then enables sharing and interoperation of data and functionality among information systems (IS). Representing the formal semantics for a certain domain implies *conceptualizing* the domain objects and their interrelationships in a declarative way. Ontologies should therefore also accommodate formal so-called *ontological commitments* (for definitions, see below) needed for new open environments such as electronic commerce, B2B, semantic web, etc. In such an open environment autonomous applications possibly developed without *a priori* knowledge about each other, still need to establish communication in order to exchange data to make transactions interoperate.

The fundamental *a-priori* shared nature of an ontology however makes it important, even essential for our understanding, to realize that ontology engineering is more

---

[1] In philosophy, Aristotle defined ontology as the science of being.

than just data modeling, even when taking business rules into account [DJM02]. Data modeling occurs within the context of a given application, and usually within one organization. Ontology building on the contrary wants to represent domain knowledge as *independently* of language and application as possible to be accessed and used by agents that have been developed autonomously. For example, representing the formal semantics of the "Car Rental" domain is more than a set of data models for a number of Car Reservation systems, which likely would have been autonomously developed for optimal use within an individual organization or company. We can say that an ontology is more *generic* than a data model. Also an ontology is more than a mere "*is_a*"- taxonomy of terms, as it often seems to appear in the literature. It typically includes a much richer set of relationships, such as *instance_of, part_of, settled_Via, Ordered_By...,* each of which might deserve its "generic semantics". Not surprisingly this turns out to be an important methodological and tool support issue! Sharing concepts, and often even just *identifying* them correctly to an application "independently" of language and across representational paradigm boundaries obviously is a hard problem of semantics. This issue has been extensively studied in various forms of *schema* and *view integration*, mostly in organization-specific contexts, within the database field ([Sa98] [AB01] [ZSC01] [PS98]).

3. SCALABILITY

Ontologies aim to be a shared and agreed semantic resource for a wide range of agents, thus, scalability is critical and a key success factor for ontology modeling approaches and ontology management systems; experience shows that "unscalable" solutions emerging from academic research often fail at the industrial level, e.g. compare the meager success of deductive database management systems with that of relational database management systems, while the former are arguably more powerful, elegant, etc.. Accordingly, we believe that the matter of scalability is more than "many-user access at runtime", it also implies the requirement of a scalable foundation (and therefore scalable methodology) for representing ontological contents itself. In short, ontology modeling must be a scalable activity, and we shall tackle this in the best model-theoretic database tradition by separating, conceptually and architecturally, the relatively stable "data" (fact) component of an ontology from its "rule" component, i.e. the more application-dependent constraints, identification, lexical representations etc. that constitute an application's so-called *commitment* to the facts.

As ontologies require an agreement about a formal semantics of a domain, generating a consensus can be a real challenge, especially for ontologies that need to span multiple domains. Several methodological aspects directly impact on scalability such as: the capability for managing the growth and evolution (versioning) of knowledge, the simplicity of the ontology building process (i.e. how easy is it to generate and manage a consensus about a domain's sememantics), and performance requirememts for storing, accessing, managing a large volumes of ontological contents.

Before building up ontological content, builders should first generate a consensus about *one* conceptualization. In [GG95] generating such a consensus is a mental process and implemented by exemplifying, testifying, investigating etc.. Another interesting approach is by a so-called Adequacy Search as proposed in [NCM+00]. Such a process will however inevitably be oriented to the *tasks* to be carried out, and are likely to be influenced also by personal tastes and may even reflect fundamental disagreements [BMC99]. Several conceptualizations could be chosen for the same domain [GN87], especially in large-scale and multi-domain ontologies, which may lead to potentially inconsistent (and incomplete) ontologies. In addition, we believe that this slows down the construction of an ontology and increases the costs.

Notice that the difficulties and disagreements in this conceptualization process normally appear at a "deeper" level of abstraction. Experiences with conceptual heterogeneity and ontology integration [GPS98] indicate that "disagreement persists at a deep, 'ontological' level …". While *constraints, rules and procedures* are essential to achieve an understanding about a domain's semantics, agreement about them in general is difficult and nearly always bound to a context of application, by expressing (restricting) how such application, typically implemented as software agents, may commit to an ontology. Furthermore, from an ontology's application point of view constraints are likely there to limit updates of data stores that exist entirely within that application's realm. Maintaining the actual consistency of those will not be the ontology's responsibility, but the application's —assumedly with the help of the ontology. For example it is easy to agree that "person has a blood-pressure", while a disagreement might be on whether the actual value of this pressure is (too) high *in a given context*. People could agree on "a book has ISBN" but might disagree whether *for a given application* that ISBN is a mandatory property for the book to have, or "person has age", but might disagree on the range.

4. KNOWLEDGE REUSABILITY

Supporting and enabling knowledge *reusability* is another important (and perhaps more immediately accessible) goal of building ontologies ([IFFJ97] [UG96] [MFGB99]

[GPB99] [G95]). Notice that ontology *usability* is subtly different from ontology reusability [RVMW99]. Increasing the reusability of knowledge implies the maximization of using this knowledge among several *kinds* of (autonomously specified!) tasks, while increasing ontology usability could mean just maximizing the number of different applications using an ontology for the same kind of task.

Similarly to the scalability issue, we claim that the capability of reuse again strongly depends on the representation and architecture of the ontological contents, i.e. on ontology models. In what follows, we discuss the activity of knowledge reuse and its relation to the knowledge level.

As a result of a conceptualization process, ontological content will stand as a formal resource of knowledge. Reusing such resources means sharing the same conceptualization. In the activity of knowledge reuse, ontologies may only need to be reused partially. For example, when building our "Car-rental" ontology, one may want to reuse the "Payment" context from an existing "Shopping" ontology, where they share the same conceptualization about a certain set of axioms. Sharing a partial conceptualization (as a result of partial agreement) across two ontologies depends on the level of abstraction, i.e. a lack of shareability appears in the "deeper" knowledge (as discussed above). To improve knowledge reusability, several researchers from the problem-solving area (e.g. Chandrasekaran and Johnson [CJ93], Clancey [C92], or Swartout and Moore [SM93]) have proposed the idea of structuring the knowledge into different levels of abstractions, while Steels in [S93] proposed a componential framework that decomposes a knowledge level into reusable components. In addition to the level of abstraction, several issues related to the reusability of knowledge are outlined and discussed in [R00] such as the importance of context, the need for more knowledge, etc.

Many believe that building large knowledge bases will only be possible if efforts are combined (Neches et al. in [PFP+92]). A unified framework to enable and maximize knowledge reusability is advisable. Such a framework becomes scalable by allowing (a) commitment to ontological contents regardless of the diversity of ontology languages and (b) simultaneous representation of "alternative worlds" as organized contexts within the ontology fact base. We implement this goal of reusability again by separating off the relatively stable declared fact base of an ontology from a layer of commitments by applications. This is reflected in the DOGMA architecture with its separate Ontology Base Server and Commitment Servers. Servicing a new task is now largely limited to this commitment layer, and "alternative worlds" are handled by representing their facts in different contexts in an ontology base.

5. THE DOGMA APPROACH TO ONTOLOGY ENGINEERING

In this section we sketch the database-inspired approach (called DOGMA) for engineering *formal ontologies*, which are implemented as shared database resources used to express agreed formal semantics for a real world domain.

5.1 Basic assumptions and formalism

We first define formal ontologies in a logic sense, i.e. as "representationless" mathematical objects that form the range of a classical interpretation mapping from a first order language (sometimes called a *conceptual schema*, and assumed to lexically represent an application), to a set of possible ("plausible") *conceptualizations* of the real world domain. We then give a database-inspired "view" on implementations of ontologies seen as resources. According to this well-tried model-theoretic database methodological principle and as indicated already above, in the DOGMA framework we decompose an ontology formally into an *ontology base*, a set of context-specific binary fact types which we call lexons (see example below), and into instances of their explicit *ontological commitments*. The latter become reified in our architecture as a separate mediating layer, see Fig.1. This also leads to methodological approaches that naturally extend database modeling theory and practice, and so may in turn lead to scalable and reusable solutions for ontology-based systems.

Example. The following ontology base contains a single —obviously very incomplete— ontology-base with lexons in a hopefully self-evident syntax, the -ID suffix denotes abstract identifiers to assumed and agreed contexts:

```
(company-ContextID)
    employee  is_a  person
    employee  is_a  contract_party
    employee  has  first_name
    employee  has  last_name
    employee  has  empl_id
    employee  has birth_date
    employee  has  start_date
    employee  has  salary
    employee  works_in  department
    {…}
(employment-ContextID)
    salary  has  amount_in-$
    salary  has  amount_in-€
    salary  synonym_of remuneration
    salary  is_a  remuneration
    salary  expressed_in  currency
    salary  converted_to  currency
{…}
```
∎

In order to maximize the "conceptual gain" of the interpretation, the formalism for specifying an ontology(-base) should be as simple as possible. Thus the ontology base is a set of conceptual relationships, with part of its formal application-specific semantics specified in the commitment layer. To accommodate alternative models of

reality, or even versions as knowledge about the world evolves e.g. through observations, the ontology base may contain many different conceptualizations, even about the "same" real world domain, organized in so-called *contexts*. The precise interpretation of contexts forms a separate research issue and the subject of another paper. In summary an ontology base is a set of possible (*plausible*) conceptualizations of the real world domain; where each is a set of context-specific binary facts types, called *lexons*. Notation: $<\gamma$: $Term_1$, role, $Term_2>$. Here $\gamma \in \Gamma$ is just an abstract *context identifier* chosen from a set, (notice that the context is not treated in this paper). The lexical terms ($Term_1$, role, $Term_2$) express a binary conceptual relationship in some given agreed language.

The commitment layer is organized as a set of *ontological commitments,* each is an explicit *instance* of an (intensional) first-order *interpretation*; each commitment is a consistent set of rules (/axioms) in a given syntax that constrains an application (or also: *commits it ontologically*) to a particular aspect of reality (which is assumed to be conceptualized in the ontology base).

Example (verbalized in a suitable pseudo-NL syntax):

   <**Each** Manager **who** Heads **a** Company **must also** Works_For **that** Company>
    ■

For improving knowledge reusability, in a commitment layer the set of ontological commitments will be seen as a set of reusable knowledge components. Such components are connected since they share the same ontology base. In practice, similar applications reuse/inherit commitments from each other. On the one hand this facilitates new applications to commit to and use the ontology, and on the other hand, successful commitments in certain domains and applications will likely become "popular" and therefore a *de facto* trusted resource in their own right.
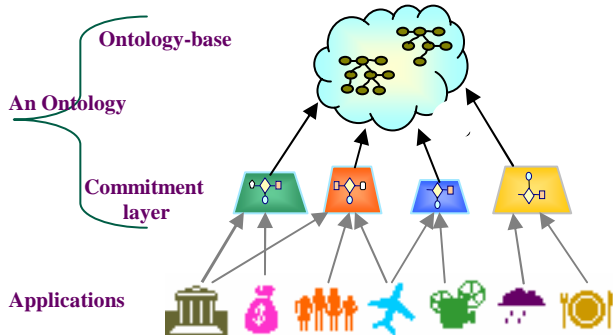

Fig. 1: Knowledge organization in DOGMA Framework

Lexons in a DOGMA ontology base are always "true",

i.e. free of further interpretation. "Alternative truths" have to be provided in separate conceptualizations or contexts.

This way of building and structuring ontologies intends to prevent application-specific rules and encodings to enter a shared ontology base. As an obvious result, building ontology bases and their commitments becomes easier and more scalable, because the rules and constraints (which mostly are the difficult part to agree) are moved to the commitment layer and the agreement about them within an ontological commitment is easier than it is within the whole ontology.

5.2 Example: A simple ontology in the DOGMA framework

The following example -with its necessary simplicity– is intended to illustrate capabilities of knowledge reusability in the DOGMA framework. Within a business domain, suppose we have a previously constructed Shopping ontology that includes Payment aspects, as labeled (both 'Shopping' and 'Payment') in fig.2. Building a new Car-Rental ontology implies the reuse of the Payment aspects, i.e. both Shopping and Car-Rental share the same conceptualization about payment[2]. In DOGMA approach, such cases should be modeled as a three ontological commitments. Fig. 2 shows (using ORM [H01]) the graphical representation of these three ontological commitments, grouped and labeled as "Shopping', "Car-Rental", and "Payment".

Notice that the ontology in this example is supposed to be specified at the knowledge level[3] i.e. is more than a data model for the application instances. Applications that commit to this ontology may retain their internal data models[4].

As discussed before, In DOGMA framework, ontology is decomposed into ontology base, as a set of lexons, and into instances of their explicit ontological commitments that form a commitment layer; respectively, Table1 and Table 2 represent the ontology base and the commitment layer of the ontology drawn in fig 2. The representation of the rules in the commitment layer is not restricted to a particular ontology language or standard, but we adopt a notational convention to specify which rules system/standard is used, in the form of a prefix of the rule. For example, the prefix

---

[2] Notice that e.g. both "shopping" and "Car-rental" ontologies may share e.g. Customer Profiling aspects, which are not presented in this example for simplicity.

[3] The Knowledge Level is a level of description of the knowledge of an agent that is independent of the symbol-level representation used internally by the agent [G95].

[4] Note that the commitments may be more than *integrity constraints* (to be committed by an application), such as derivation or reasoning rules that may help to enrich or filter queries.

"ORM." is used in Table 2 for rules that are intended to be interpreted as "standard" ORM ([H01]) by "standard ORM" tools. Furthermore, as *ontological view*, each ontological commitment should define which lexons are used and constrained in that particular commitment. E.g., for simplicity we allow the use of rules number e.g. 1, 4, and 8 to show that the symbolic representation of those lexons will be visible as they are defined in the ontology base. Notice that we present the ORM rules in Table 2 by verbalizing them into fixed-syntax English sentences (i.e. generated from agreed templates parameterized over the ontology base content). We believe that this should allow non-experts to (help to) check, validate or build the commitment rules and will simplify the commitment

modeling process. For ORM, verbalizations may eventually be replaced by RIDL Constraint Language expressions ([VB82], [DMV88]) or expressed in another formalism, and in such case we may compile them (RIDL-A, [DMV88]).

As a design pattern in DOGMA framework, when building an ontology from scratch, builders may decide to model the whole ontology into one ontological commitment, which is possible but not encouraged in DOGMA framework; so, ontology builders should split their knowledge into a considerable number of ontological commitments, taking into account the maximization of knowledge reusability.
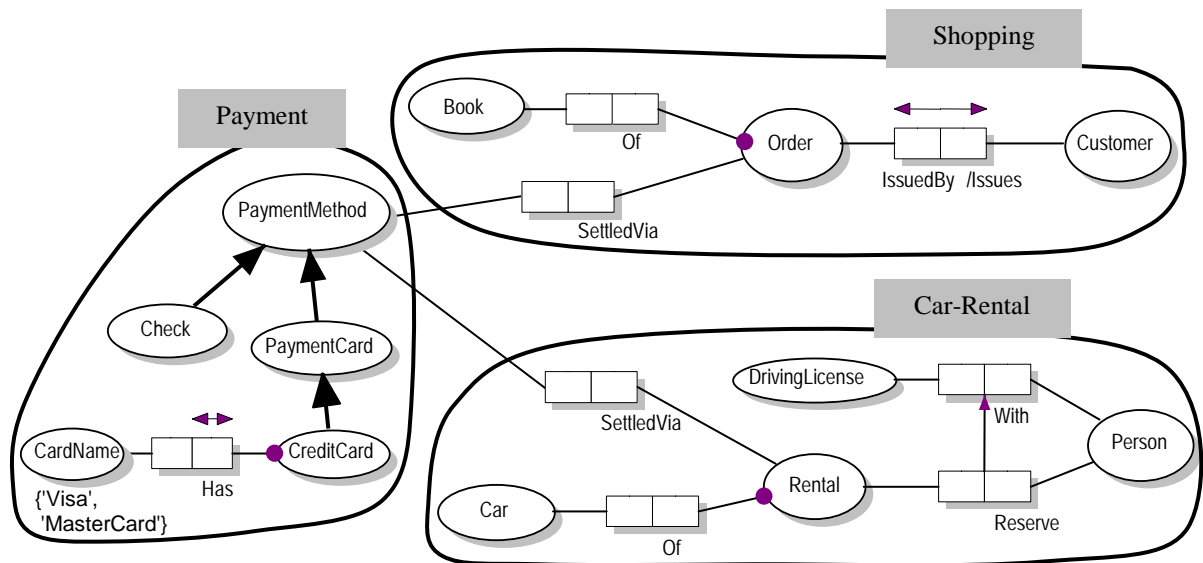

Fig. 2: Business ontology

LNo	Context	Term1	Role	Term2
		Ontology base (Lexons)		
1	Business	Customer	Issues	Order
2	Business	Order	IssuedBy	Customer
3	Business	Order	Of	Book
4	Business	Order	SettledBt	PaymentMethod
5	Business	PaymentCard	IsA	PaymentMethod
6	Business	Check	IsA	PaymentMethod
7	Business	CreditCard	IsA	PaymentCard
8	Business	CreditCard	Has	CardName
9	Business	Visa	IsInstanceOf	CardName
10	Business	MasterCard	IsInstanceOf	CardName
11	Business	Person	With	DrivingLicense
12	Business	Person	Reserve	Rental
13	Business	Rental	Of	Book
14	Business	Rental	SettledBy	PaymentMethod

Table 1: The Ontology Base

RuleID	Rule Definition	Commitment
1	DOGMA. Visible Lexons to this commitment are {$1 .. $4}	Shopping
2	ORM.Mandatory(Each Order Of at least one Book)	Shopping
3	ORM.InternalUniqueness(It is disallowed that the same Customer Issues the same Order more than once, and it is disallowed that the same Order IssuedBy the same Customer more than once)	Shopping
4	DOGMA. Visible Lexons to this commitment are {$5 .. $10}	Payment
5	ORM.Mandatory(Each CreditCard Has at least one CardName)	Payment
6	ORM.InternalUniqueness(Each CreditCard Has at most one CardName)	Payment
7	ORM.Value (The possible values of CardName are: 'Visa', 'MasterCard')	Payment
8	DOGMA. Visible Lexons to this commitment are {$11 .. $14}	Car-Rental
9	ORM.SubSet(Each Person who Reserve a Car must be With DrivingLicense)	Car-Rental
10	ORM.Mandatory(Each Rental Of at least one Car)	Car-Rental

Table 2: The Commitment layer
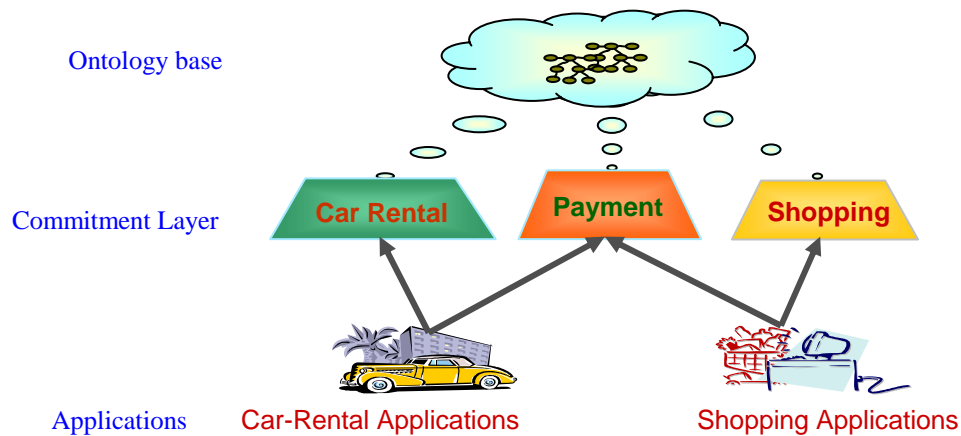
Fig. 3: DOGMA's Knowledge Organization of example 5.1

As a result, structuring the knowledge into two levels of abstraction and organizing them into an ontology base and its set of ontological commitments will ultimately increase the knowledge reusability for several kinds of tasks. Fig. 3 shows that shopping applications use two commitments (Shopping, Payment), while Car-Rental applications use the commitments (Car-Rental, Payment).

5.3 Discussion of the formalism and architecture

From the previous example we can clearly see that DOGMA approach and architecture improve scalability, as the simultaneous representation and organization of multi-domain (normally large-volume) ontologies is possible; in addition, the simplicity of storing the ontology base and the commitment layer in a high performance DBMS will certainly speed up the multi-user access and retrieve facilities. Moreover, building ontology bases and their commitments becomes easier and more scalable, because the rules and constraints (which mostly are the difficult part to agree) are moved to the commitment layer, as discussed before.

While ontologies are being engineered they grow (and are modified) over time or domain. Therefore *versioning mechanisms* normally adopted to deal with changes may cause consistency problems for the applications that commit to the ontology, as noted already in [KF01]. Adopting our approach, the need for an ontology versioning mechanism is simplified: (a) lexons can be added to the ontology base without any effect to the ontological commitments; and (b) lexons cannot be deleted or modified if they are in use (see rules 1, 4 and 8 in Table2). Adding or modifying rules in the ontological commitments also becomes easier to manage for a versioning mechanism, as the number of applications

committing to a given ontological commitment in general is less than those committing to the whole ontology, therefore again reducing the impact of changes to be controlled.

6. CONCLUSION

In this paper we discussed the scalability and knowledge reusability principles in ontology modeling, then we illustrated how the DOGMA framework improved these principles by structuring an ontology, conceptually and architecturally, as an ontology base (relatively stable "data" (fact) component) and commitment layer (its "rule" component).

ACKNOWLEDGMENT

REFERENCES

[AB01]   Alagic, S. and Bernstein, P.A.: "A Model Theory for Generic Schema Management". In Proceedings of DBPL'01, G. Ghelli and G. Grahne (eds), Springer-Verlag, (2001).

[BCM99]   Bench-Capon T.J.M., Malcolm G., "Formalising Ontologies and Their Relations". In Proceedings of DEXA'99: 250-259, (1999).

[CJ93]   Chandrasekaran B., and Johnson T.R., "Generic Tasks and Task Structures: History, Critique and New Directions," in Second Generation Expert Systems, David J.M., Krivine J.P., and Simmons R. (Eds.), Springer-Verlag, London, pp. 233-272. (1993).

[C92]   Clancey W.J., "Model construction operators". *Artificial Intelligence*, 53(1):1-115, (1992).

[DJM02]   Jan Demey, Mustafa Jarrar, Robert Meersman: "Markup Language for ORM Business Rules", In International Workshop on Rule Markup Languages for Business Rules on the Semantic Web, First International Semantic Web Conference, ISWC, (2002).

[DMV88]   De Troyer, O.M.F., Meersman, R. and Verlinden, P., "RIDL* on the CRIS Case: A Workbench for NIAM". In Computerized Assistance during the Information Systems Life Cycle, T.W. Olle, A.A. Verrijn-Stuart, and L. Bhabuta (eds), pp 375--459, North-Holland/IFIP Amsterdam, (1988).

[G95]   Gruber T.R., "Toward principles for the design of ontologies used for knowledge sharing", International Journal of Human-Computer Studies, 43(5/6), (1995).

[GG95]   Guarino, N. and Giaretta, P., "Ontologies and Knowledge Bases: Towards a Terminological Clarification"` in: Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing, N. Mars (ed.), pp 25-32, IOS Press, Amsterdam (1995).

[GPB99]   Gomez-Perez, A., & Benjamins, R., "Overview of Knowledge Sharing and Reuse Components: Ontologies and Problem-Solving Methods", in: Proceedings of the IJCAI-99 Workshop on Ontologies and Problem-Solving Methods (KRR5), Morgan-Kaufmann (1999).

[GPS98]   Gangemi A., Pisanelli D., and Steve G., "Ontology Integration: Experiences with Medical Terminologies", in: Formal Ontology in Information Systems, N. Guarino (ed.), IOS Press (1998).

[GN87]   Genesereth, M.R. and Nilsson, N.J., Logical Foundation of Artificial Intelligence. Morgan Kaufmann, Los Altos, California (1987).

[H01]   Halpin, T. "Information Modeling and Relational Databases", 3rd ed., Morgan-Kaufmann, (2001).

[IFFJ97]   Iwasaki I., Farquhar A., Fikes, R., and Rice J., "A Web-Based Compositional Modeling System for Sharing of Physical Knowledge", in: Proceedings of IJCAI'97 Conference, pp. 494-500. Morgan-Kaufmann (1997)

[KF01]   Klein, M. and Fensel, D. "Ontology Versioning on the Semantic Web", in: First International Semantic Web Working Symposium (SWWS-1) (2001).

[MFGB99] Motta, E., Fensel, D., Gaspari, M. and Benjamins, R. Specifications of Knowledge Components for Reuse, In Proceedings of the 11th International Conference on Software Engineering and Knowledge Engineering, Kaiserslautern, Germany, KSI Press, pp. 36-43 (1999).

[NCM+00]Da Nobrega G.M., Castro E., Malbos P., Sallantin J., Cerri S.A., "A framework for supervised conceptualizing", In ECAI'00: pp. 17.1-17.4. (2000).

[PFP+92] Patil R.S., Fikes R.E., Patel-Schneider P.F., McKay D., Finin T., Gruber T., Neches R., "The DARPA Knowledge Sharing Effort: Progress Report", in: Proc. of Knowledge Representation and Reasoning, 777-788, (1992).

[PS98]   Parent C., Spaccapietra, S.: "Database Integration: an Overview of Issues and Approaches", CACM, vol. 41, no 5, pp. 166-178, May (1998).

[R00]   Richards. D. "The Reuse of Knowledge: A User-Centered Approach", *International Journal of Human Computer Studies*, (2000).

[RVMW99] Thomas Russ, Andre Valente, Robert MacGregor & William Swartout, "Practical Experiences in Trading Off Ontology Usability and Reusabilityhttp ://sern.ucalgary.ca/KSI/KAW/KAW99/papers/Russ1/kaw9 9-4-14.pdf",   In Twelfth Workshop on Knowledge Acquisition, Modeling and Management, (1999).

[S93]   Steels, L., "The componential framework and its role in reusability", in: Second Generation Expert Systems, J.-M. David, J.-P. Krivine & R.Simmons, (eds.), pp. 273-298. Berlin: Springer-Verlag (1993).

[Sa98]   Santucci, G., "Semantic Schema Refinements for Multilevel Schema Integration", *Data and Knowledge Engineering* 25 (1998), pp. 301-326, North-Holland.

[SGP98]   Steve G., Gangemi A., Pisanelli D.M., "Integrating Medical Terminologies with the ONIONS Methodology", in: Information Modeling and Knowledge Bases VIII, Kangassalo H., Charrel J.P. (eds.), IOS Press Amsterdam (1998).

[SM93]   Swartout, W.R. and Moore, J.D., "Explanation in Second Generation Expert Systems", in: Second Generation Expert Systems, J.-M. David, J.-P. Krivine and R. Simmons (eds),. Berlin: Springer-Verlag (1993).

[UG96]   Uschold, M. and Gruninger, M, "Ontologies: principles, methods and applications", *Knowledge Engineering Review*, vol. 11, no. 2 (1996).

[VB82]   Verheyen, G. and van Bekkum, P.: NIAM, aN Information Analysis Method. In: IFIP Conference on Comparative Review of Information Systems Methodologies, T.W. Olle, H. Sol, and A. Verrijn-Stuart (eds.), North-Holland (1982).

[ZSC01]   Yannis Tzitzikas, Nicolas Spyratos and Panos Constantopoulos, "Mediators over Ontology-based Information Sources", in: Second International Conference on Web Information Systems Engineering (WISE 2001), Kyoto, Japan, (2001).