From Visual Language to Ontology Representation: Using OWL for Transitivity Analysis in 4EM

Birger Lantow, Kurt Sandkuhl

University of Rostock, Chair of Business Information Systems,
18051 Rostock, Germany,
E-Mail: birger.lantow@uni-rostock.de

Abstract. Usually, enterprise models consider different aspects and include different abstraction levels of enterprises. The application of ontologies as conceptual bases that can clarify relations within and between these abstraction levels is believed to be helpful. This paper investigates the use of ontologies for formalizing enterprise modelling languages and enriching their semantics. The aim is to transform enterprise models into ontologies based on a mapping of the enterprise models' meta-model into a semantically corresponding ontology. The ontology representation then is used to check logical consistency and to infer new facts regarding the implications of the model beyond what would be possible with a visual modelling language. In order to check feasibility and pertinence of our approach, we selected the goal modelling part of the 4EM method. This paper provides (1) a formal OWL representation of the 4EM goals meta-model; (2) a systematization of transitive goal properties; (3) a set of SWRL rules expressing these transitivity; and (4) an analysis of exemplary goals model instances.

Keywords: 4EM, OWL, Enterprise Architecture, Enterprise Modelling, Goal Modelling, SWRL, Enterprise Model Analysis, Meta-Modelling.

1 Introduction

In general terms, enterprise modelling is addressing the systematic analysis and modelling of processes, organization structures, products, IT-systems or other perspectives relevant for the modelling purpose [1]. Usually, enterprise models consider different enterprise aspects and include different abstraction levels induced by refinements of, e.g., processes into sub-processes or goals into sub-goals. Ontologies are content theories about the sorts of objects, properties of objects and relations between objects possible in a specified knowledge domain [2]. The application of ontologies as conceptual bases that can clarify relations within and between different abstraction levels in enterprise models is believed to be helpful. Ontologies have shown their usability for this type of tasks. They provide a way of knowledge representation, which is widely used today for intelligent analysis of

S. España J. Ralyté, P. Soffer, J. Zdravkovic and Ó. Pastor (Eds.):
PoEM 2015 Short and Doctoral Consortium Papers, pp. 51-60, 2015.

knowledge. As a consequence of this, ontologies will also have the power to clarify the relations between focal areas and the constructs within a focal area [3].

This paper investigates the use of ontologies for formalizing enterprise modelling languages and enriching their semantics. The focus in this context is on visual languages which have the advantage to be better understandable by non-experts in enterprises but which in most cases lack operational semantics (see [4] for an overview). More concrete, we aim at transforming enterprise models into ontologies based on a mapping of the enterprise models' meta-model into a semantically corresponding ontology. From the existing ontology representations, we will use the W3C recommendation ontology language OWL (Web Ontology language) in its version OWL2 to represent the ontology. An OWL ontology consists of Individuals, Properties and Classes.

The ontology representation then is used to check logical consistency and to infer new facts regarding the implications of the model beyond what would be possible with a visual modelling language. In order to check feasibility and pertinence of our approach, we selected one modelling language (4EM; see section 2) and focused within 4EM on the goal modelling part.

This paper provides (1) a formal OWL representation of the 4EM Goals meta-model; (2) a systematization of transitive goal properties; (3) a set of SWRL rules expressing this transitivity; and (4) an analysis of exemplary goals model instances.

The remainder of the paper is structured as follows: Section 2 describes the construction of an ontology representing the 4EM meta-model for goal modelling. Section 3 shows how this ontology can be enriched by adding transitivity rules. Section 4 provides a validation of the formalized meta-model by instantiating an example model coming with the 4EM specification in [5]. Section 5 summarizes the work and discusses future activities.

2 4EM Goal Modelling Ontology

Experience reports on Enterprise Modelling indicate both, the usefulness of ontology representations [6] and the inclusion of goals [7]. Ontologies have been used for many years for representing enterprise models. The most popular examples are probably Uschold et al.'s "The Enterprise Ontology" [8] and Dietz's DEMO approach [9]. Although the Enterprise Ontology aims at representing business objectives, an appropriate concept structure for representing goal relations is not available. In DEMO, goals could be represented by using the "agendum" concept, but this concept has a wider meaning than just goals. The DIO ontology provides representation of the ArchiMate meta-model [10]. ArchiMate's motivation extension allows for the representation of goals. However, structured goal hierarchies and relations for other perspectives in enterprise models are not developed in ArchiMate to the same extent as in 4EM.

From the existing EM methods, the „For Enterprise Modelling (4EM)" [4] has been selected for this paper because of the expressive goal modelling possibilities and the publicly available documentation including an informal meta-model. 4EM uses six interrelated sub-models which complement each other and capture different views of the enterprise, i.e. each of the sub-models represents some aspect of the enterprise.

These sub-models are: (1) Goals Model, (2) Business Rule Model , (3) Concepts Model, (4) Business Processes Model, (5) Actors and Resources Model, and (6) Technical Components Model.

The Goals Model focuses on describing the goals of the enterprise. This model captures what the enterprise and its employees want to achieve, or to avoid. Goals Models usually clarify questions, such as:

- Are there conflict/support relationships between goals?
- Are there constraints/problems that hinder the achievement of a goal?
- What sub-goals have to be achieved in order to achieve a goal?
- What generally hinders/supports the achievement of a goal?

These so called Competency Questions (QC) can be used as a requirements specification for an ontology on the domain of enterprise goals [11].

Especially in complex models visual analysis of these aspects is error prone. If a sub-goal is in a conflict or underlies some constraints, these circumstances should also be considered at top-level. Furthermore, inherent inconsistencies like supporting top-level goals having conflicting sub-goals need attention. Ontology-based reasoning provides a tool to assess these issues stemming from transitive relationships in goal modelling.

In the following, the ontological representation of the 4EM Goals meta-model will be constructed according to the 4EM method description in [5, pp. 87-101]. First the taxonomy of goals model component types (classes) is constructed. In a second step, the construction of binary and n-ary relationship types follows. Relationship transitivity is discussed separately in section 3 because it is not specified in [5]

2.1 Goal Model Component Types

The model component types are represented as classes in OWL. All goals model component types are represented as specializations of the abstract class `GM_ModellingComponent`. The `Goal` class represents goals or objectives respectively. The 4EM method describes priority and criticality as optional attributes for goals. These have not been considered in the meta-model so far. This is kept for later work. Problems symbolize environmental circumstances that hinder the achievement of goals. Problems can be described more specifically as weaknesses (internal factors) and threats (external factors. Problems are represented in OWL with the `Goal` class and its sub-classes `Threat` and `Weakness`. A cause expresses explanations or reasons for problems (`Cause` class). Apart from causes, constraints (`Constraint` class) express business restrictions, laws or external policies that affect components of the goals model. The last component type are opportunities (`Opportunity` class) which symbolize resources supporting the achievement of certain goals.

2.2 Goal Model Binary Relationship Types

Relationship types are represented as object properties in OWL. Object properties are directed binary relationships. Further semantics can be added to object properties by defining characteristics like transitivity and relations to other object properties, including specialization/generalization.

The 4EM goals model describes four binary relationship types. First, the `supports` relationship shows that fulfilling one goal also supports the achievement of another. Furthermore, the relationship is used to relate opportunities to goals. The `contradicts` relationship in contrast shows that the achievement of one goal is in conflict with another. This relationship is considered to be symmetric. Hence, if goal A contradicts goal B also goal B contradicts goal A. The `hinders` relationship is less strict. It can be used between model components to show negative influences. This relationship is not considered symmetric but can also be used to link goals. The last binary relationship is the `causes` relationship. It is used to link causes to problems.

Experience from ontology engineering shows that inverse relationships should be included in an ontology in order to fully specify concept relationships. For example, a problem can be linked to one of its causes by a `caused_by` relationship. These inverse relationships are automatically added to instances by OWL reasoning if defined in the meta-model. Table 1 shows the specification of the binary relationships.

Table 1. Goals Model Object Properties

Object Property	Domain	Range	Inverse	Characteristics
supports	Supporter	Goal	supported_by	transitive
contradicts	Goal	Goal	-	symmetric
hinders	Hinderer	Goal	hindered_by	-
causes	Cause	Problem	caused_by	-

Two additional abstract classes have been added. `Supporter` for goals model element types that can support the achievement of a goal (sub-classes `Goal` and Opportunity) and `Hinderer` for element types that can have a negative influence on the achievement of a goal (sub-classes `Goal`, `Problem`, and `Constraint`). The supports relationship is considered to be transitive. Hence, if A supports B and B supports C, A also supports C. Similar assumptions cannot be made for the other binary relationships.

2.3 Goal Model N-ary Relationship Types

N-ary relationships define semantics of goal decomposition in the 4EM goals model. The AND-relationship decomposes a top-goal into a set of sub-goals that have to be fulfilled each in order to achieve the top-goal. The OR-relationship defines a set of sub-goals where it is sufficient to fulfill one of the alternatives. Finally the AND/OR-relationship needs a combination of some of the sub-goals to be fulfilled. N-ary relationships are not directly supported in OWL. Logical Ontology Design Patterns can be used in order to model cases where the ontology language does not provide appropriate constructs [12]. The catalogue of the NeON-projects provides the n-ary relationship pattern for modelling such relationship types in OWL [13]. A class for the relationship type is created and appropriate object properties are associated. For goals modelling, the abstract class `GoalComposition` is used to represent decomposition of goals. The respective sub-classes are `ANDGoals`, `ORGoals`, and

`ANDORGoals`. Accordingly, object properties have been defined. The `compositionTopGoal` property assigns the goal to be decomposed and the `compositionSubGoal` property assigns the sub-goals. `topGoalComposedBy` and `SubGoalComposedIn` are the respective inverse properties. According to the 4EM method, goal composition structures are special cases of the supports relationship. Therefore, the chain of composition object properties is defined as a sub-property of `supports` (`subGoalComposedIn o compositionTopGoal SubPropertyOf supports`). Fig. 1 shows the complete OWL class hierarchy that is used to represent the 4EM goals meta-model.

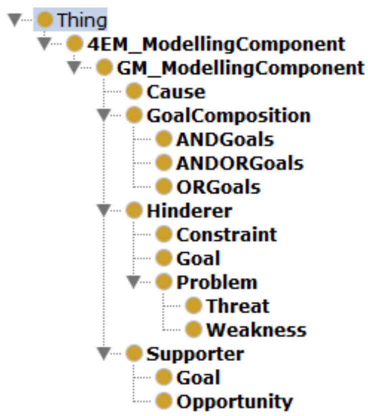

Fig. 1. Extended Class Hierarchy of the 4EM Goals Meta-Model

3 Transitivity Rules

After modelling the 4EM goals meta-model, the possibility of formal statements regarding transitivity of goal properties is investigated. In a first step a systematic analysis of possible property propagations between goals is performed (section 3.1) . In a second step, the formalization of found transitivity is discussed (section 3.2).

3.1 Transitivity of Goal Properties

By systematically investigating transitivity we analyse which object properties are shared between goals based on the possible goal-to-goal relationships. In addition, we also ask which object properties may be assumed for a goal at the target of a goal-to-goal relationship based on the object properties of the goal at the origin. Table 2 shows all possible combinations and the assumptions made for transitivity. The columns contain the goal-to-goal relationships. Considering the direction of these relationships, the direction of property propagation is set. Relationship semantics do not allow property propagation along the inverse relationships defined in section 2.

For example, if goal A is hindered by some hinderer H and supported by goal B no assumptions can be made for the relationship between B and H.

The rows contain the object properties to be propagated. Referring to the discussion in section 2, these include the relationships originally defined by the 4EM method and their inverse properties as well. For example, a goal can hinder another goal and can be hindered by some Hinderer as well.

A first decision made for transitivity specification is the exclusion of property propagation via goal conflicts and hinders relationships. For example, if goal B hinders goal A and goal B is hindered by goal C no assumption can be made that goal C supports goal A (double negation). The same applies for the contradiction relationship. Influences like constraints and problems that pose difficulties for the achievement of a goal may not influence the achievement of another goal at all or may also have a negative influence on it. Furthermore, goals are desired future states. Conflicts between goals need to be solved by a decision in favor of one of the goals or by relating the degree of goal fulfillment. The focus should be on the goals not on relating the context of one goal to the other.

The situation is different for supports relationships between goals. The semantics of these relationships means that a sub-goal is a more specified part of the top-goal (cf. [5]). Thus, the context of the sub-goals is also part of the top-goals' context. This is also true for goal compositions. As described in section 2, goal compositions form specializations of the supports relationship. Thus, their semantics are generally the same. This is also true for object property propagation. However, the AND-composition requiring all sub-goals to be fulfilled allows for the definition of more strict (specialized) semantics for object property propagation. In consequence, table 3 has just two columns: for the supports relationship and for the AND-composition.

Propagating hinders, supports and contradicts via supports relationships is not considered. In contrast to the AND composition, the sub-goal is not required to be fulfilled in order to achieve the supported top-goal. For example, if goal A supports goal B and goal A hinders goal C, it cannot be concluded that the fulfillment of goal A also hinders goal C.

Table 2. Object Property Transitivity by Goal-to-Goal Relationships

	supports	**AND composed in**
hindered by	hindered by	hindered by
supported by	supported by	supported by
contradicts	hindered by	contradicts
AND composed by	supported by	AND composed by
OR composed by	supported by	supported by
AND/OR composed by	supported by	supported by
hinders	-	hinders
supports	-	supports
contradicts	-	contradicts

3.2 Formalization

After clarifying which object property propagation semantics should be supported, a formalization of these semantics is required. Generally, there are two possibilities to add such object property related semantics for inference mechanisms. First, the OWL language can be used. Here, object property axioms provide means to infer object property assertions (relationships between instances) based on existing object property assertions. Second, a rule language like SWRL can be used. Here, new facts are inferred based on a test of freely defined OWL statements against the ontology. If the body of a rule is found to be true its head is considered true as well and a new fact can be added to the ontology.

Transitivity of the supports relationship has already been defined in section 2 and can be expressed in OWL. However, property chains as introduced in section 2 are not fully supported by current OWL reasoning tools (Hermit 1.3.8). Thus, only SWRL rules are used to address the transitivity along property chains and n-ary relations. Table 4 shows the resulting formalization of the transitivity rules discussed in section 3.1. An ontology containing the instances of the example from section 4 can be found here: http://win.informatik.uni-rostock.de/uploads/media/4EM_GM.owl

Table 3. OWL/SWRL Formalization of Object Property Transitivity

<table>
<tr><th colspan="2">supports</th></tr>
<tr><td>hindered by</td><td><code>hinders(?c,?SubGoal),supports(?SubGoal,?TopGoal) -></code>
<code>hinders(?c,?TopGoal)</code></td></tr>
<tr><td>supported by</td><td><code>supports</code> is defined transitive</td></tr>
<tr><td>contradicts</td><td><code>supports(?SubGoal, ?TopGoal), contradicts(?h, ?SubGoal)</code>
<code>-> hinders(?h, ?TopGoal)</code></td></tr>
<tr><td>AND composed by</td><td rowspan="2"><code>subGoalComposedIn(?SubGoal,?Comp),</code>
<code>compositionTopGoal(?Comp,?TopGoal) -></code>
<code>supports(?SubGoal,?TopGoal)</code></td></tr>
<tr><td>OR composed by</td></tr>
<tr><td>AND/OR composed by</td><td><code>supports</code> is defined transitive</td></tr>
<tr><th colspan="2">AND composed in</th></tr>
<tr><td>hindered by</td><td><code>subGoalComposedIn(?SubGoal,?Comp),</code>
<code>compositionTopGoal(?Comp,?TopGoal) -></code>
<code>supports(?SubGoal,?TopGoal)</code>
<code>hinders(?c,?SubGoal),supports(?SubGoal,?TopGoal) -></code>
<code>hinders(?c,?TopGoal)</code></td></tr>
<tr><td>supported by</td><td><code>subGoalComposedIn(?SubGoal,?Comp),</code>
<code>compositionTopGoal(?Comp,?TopGoal) -></code>
<code>supports(?SubGoal,?TopGoal)</code>
<code>supports</code> is defined transitive</td></tr>
<tr><td>contradicts</td><td><code>ANDGoals(?ANDComp), compositionSubGoal(?ANDComp, ?SubGoal),</code>
<code>compositionTopGoal(?ANDComp,?TopGoal),contradicts(?c,?SubGoal)</code>
<code>-> contradicts(?c, ?TopGoal)</code></td></tr>
<tr><td>AND composed by</td><td><code>ANDGoals(?ANDComp), compositionSubGoal(?ANDComp, ?SubGoal),</code>
<code>compositionTopGoal(?ANDComp,?TopGoal),</code>
<code>ANDGoals(?ANDSubComp), compositionSubGoal(?ANDSubComp,</code>
<code>?SubSubGoal), compositionTopGoal(?ANDSubComp,?SubGoal)</code>
<code>-> compositionSubGoal(?ANDComp, ?SubSubGoal)</code></td></tr>
<tr><td>OR composed by</td><td rowspan="2"><code>subGoalComposedIn(?SubGoal,?Comp),</code>
<code>compositionTopGoal(?Comp,?TopGoal) -></code>
<code>supports(?SubGoal,?TopGoal)</code></td></tr>
<tr><td>AND/OR</td></tr>
</table>

composed by	`supports` is defined transitive
hinders	`ANDGoals(?ANDComp), compositionSubGoal(?ANDComp, ?SubGoal), compositionTopGoal(?ANDComp, ?TopGoal), hinders(?SubGoal, ?c) -> hinders(?TopGoal, ?c)`
supports	`ANDGoals(?ANDComp), compositionSubGoal(?ANDComp, ?SubGoal), compositionTopGoal(?ANDComp, ?TopGoal), supports(?SubGoal,?c) -> supports(?TopGoal, ?c)`
contradicts	`contradicts` is defined symmetric

4 Exemplary Model Analysis

In order to assess the applicability of the ontology to 4EM Goals models and the benefits of OWL reasoning, we have adopted the exemplary A4Y case from [5]. Some minor changes have been made in order to add complexity and to simulate a less systematic modelling. The hinders relationship between Goal 2 and 3 in [5] has been removed in favor of a sub-goal (Goal 10) of Goal 2 hindering Goal 3. Furthermore, Goal 10 has been split into 2 two goals (9 and 10).

It was possible to instantiate the complete model using the ontology. Furthermore, inference with the Hermit 1.3.8 reasoner added new facts to the model. No logical errors have been found. Regarding the relationship between Goal2 and Goal 3 it was inferred that Goal 2 hinders Goal 3 because Goal 10 is necessary to achieve Goal 2 and hinders Goal 3 at the same time (see Fig. 3). Thus, even if those hidden relationships are not modelled directly they reveal by reasoning using the proposed ontology schema and rules. Furthermore, the complete context is constructed automatically for a goal. All hindering and supporting influences are assigned to the goals for detailed analysis. Thus the Competency Questions formulated in section 2 can be answered by the ontology. The ontology allows for inferring hidden contradictions and hinders relations as described for the case of Goal 2 and Goal 3. These could be missed when relying on visual analysis only. Additionally, ontology based queries can be performed for further analysis. For example, goals that have hinders and supports relationships to each other at the same time need special attention and can be identified (G2 supports and hinders G1 in Fig. 3). Reasons may be conflicting sub-goals.

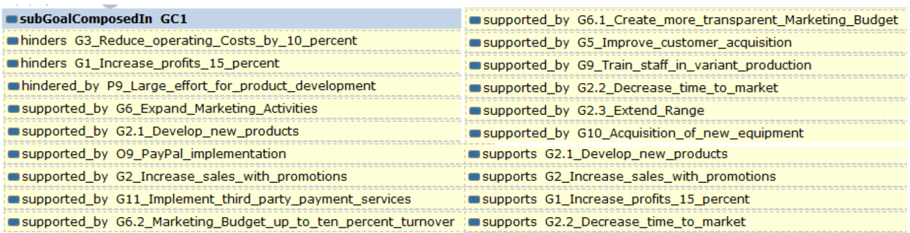

Fig. 2. Inferred Relations for Goal 2 "Increase Sales with Promotions"

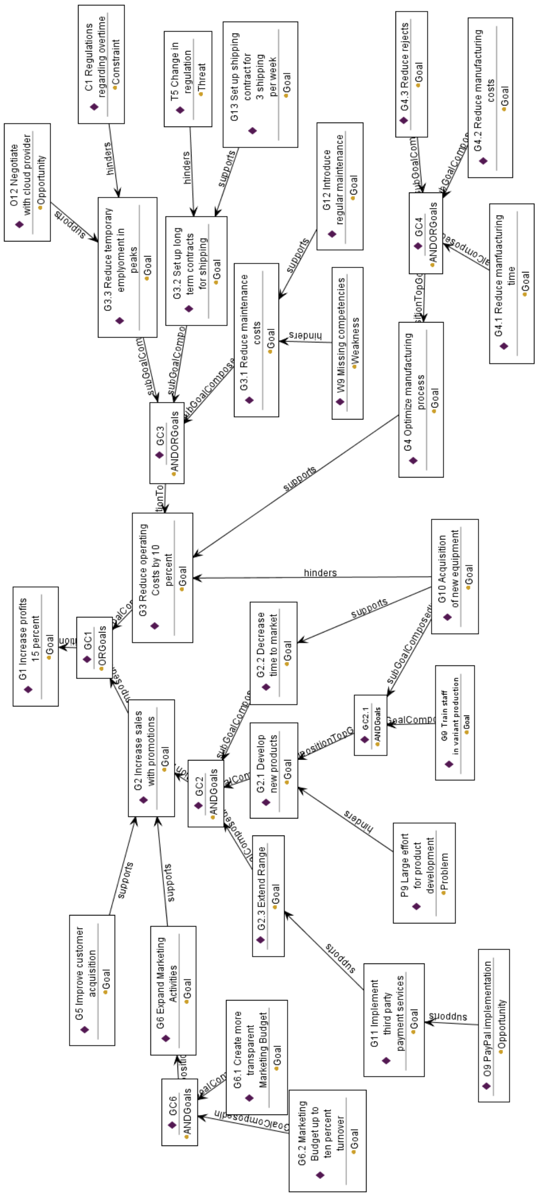

Fig. 3. Exemplary Goals Model Instance

5 Summary and Outlook

Based on the example of 4EM goal modelling, this paper investigated the possibility to transform meta-models of existing enterprise modelling languages into ontologies.

The purpose of this transformation was to further specify the relations between focal areas and the constructs within a focal area, to check logical consistency, and to infer new facts regarding the implications of the model beyond what would be possible with a visual modelling language.
Our work showed that the developed ontology is applicable and the implemented reasoning provides support for analysis of the goal model.

Future work will have to investigate the implications of an ontology-based formalization for 4EM and the transferability of results to other enterprise modelling languages. In order to understand the implications for 4EM we started to capture the complete meta-model of 4EM in an ontology, i.e. to extend the goal modelling ontology to all perspectives of 4EM. This overall 4EM ontology will have to be used to check inconsistencies and clarification needs in 4EM. We expect that more transitivity rules and reverse relationships will have to be added. Regarding transferability to other enterprise modelling languages, we do not expect general problems as long as the language in question does not define operational semantics.

References

1. Vernadat, F. B. (1996). Enterprise Modelling and Integration. Chapman & Hall, 1996.
2. Chandrasekaran B., Josephson, J. R. and Benjamins, V. R., 1999. What are ontologies and why do we need them? IEEE Intelligent Systems, Jan/Feb, 14(1), pp. 20-26.
3. Kaczmarek, T., Seigerroth, U., Shilov, N., 2012. Multi-layered enterprise modeling and its challenges in business and IT alignment, Proceedings ICEIS 2012, Wroclaw, Poland, June 28 – July 01, 2012, pp. 257-260.
4. Sandkuhl, K.; Stirna, J.; Persson, A. and M. Wißotzki (2014) Enterprise Modeling: Tackling Business Challenges with the 4EM Method (The Enterprise Engineering Series). Springer Verlag, Berlin Heidelberg. ISBN 978-3662437247.
5. Sandkuhl, K., Stirna, J., Persson, A., & Wißotzki, M. (2014). Enterprise modeling: Tackling business challenges with the 4EM method. Springer.
6. Sandkuhl, K., Smirnov, A., Shilov, N., & Koç, H. (2015, June). Ontology-Driven Enterprise Modelling in Practice: Experiences from Industrial Cases. In Advanced Information Systems Engineering Workshops (pp. 209-220). Springer International Publishing.
7. Kavakli, V., & Loucopoulos, P. (1999). Goal-driven business process analysis application in electricity deregulation. Information Systems, 24(3), 187-207.
8. Uschold, M., King, M., Moralee, S., & Zorgios, Y. (1998). The enterprise ontology. *The knowledge engineering review*, *13*(01), 31-89.
9. Dietz, J. (2006) Enterprise Ontology: Theory and Methodology. Springer.
10. Bakhshandeh, M.; Antunes, G.; Mayer, R.; Borbinha, J.; Caetano, A., "A Modular Ontology for the Enterprise Architecture Domain," Enterprise Distributed Object Computing Conference Workshops (EDOCW), 2013 17th IEEE International , vol., no., pp.5,12, 9-13 Sept. 2013
11. Noy, N. F., & McGuinness, D. L. (2001). Ontology development 101: A guide to creating your first ontology.
12. Gangemi, A. and V. Presutti (2009) Ontology design patterns. In Handbook on Ontologies, 2nd Ed., International Handbooks on Information Systems. Springer, 2009.
13. EU-FP7 funded IP NeON: http://www.neon-project.org