

A Metamodel for Supporting Interoperability in Heterogeneous Ontology Networks

Rodrigo Bonacin^{1,2}, Ivo Calado³, and Julio Cesar dos Reis⁴

¹ UNIFACCAMP, Rua Guatemala, 167, 13231-230, Campo Limpo Paulista-SP, Brazil

² CTI, Rodovia Dom Pedro I, km 143,6, 13069-901, Campinas-SP, Brazil
rodrigo.bonacin@cti.gov.br

³ Federal Institute of Alagoas, 57100-000, Rio Largo-AL, Brazil
ivo.calado@ifal.edu.br

⁴ Institute of Computing, UNICAMP, Campinas-SP, Brazil
jreis@ic.unicamp.br

Abstract. Ontologies are central artifacts in modern information systems. Ontology networks consider the coexistence of different ontology models in the same conceptual space. It is relevant that computational systems specified with distinct models based on different methods, as well as divergent metaphysical assumptions, exchange data to interoperate one with the other. However, there is a lack of techniques to enable the adequate conciliation among models. In this paper, we propose and formalize a metamodel to enable the construction of data models aiming to support the interoperability at the technical level. We present the use of our metamodel to conciliate, without explicit transformations, Ontology Charts from Organizational Semiotics with Semantic Web OWL ontologies and less structured models such as soft ontologies. Our results indicate the possibility of identifying an entity from one model into another, enabling data exchange and interpretation in heterogeneous ontology network.

Keywords: Ontology Chart, OWL ontologies, Soft ontologies, Metamodeling.

1 Introduction

The adopted solutions for expressing and representing meanings in information systems differ according to various contexts, needs, and design approaches. The term ontology in Computer Science is often used to refer to the semantic understanding (a conceptual framework of knowledge) shared by individuals participating in a given knowledge domain. More specifically, an ontology is used to formally specify the concepts and relationships that characterize a certain body of knowledge (domain) [9]. The Semantic Web ontologies provide means for data semantic interpretation and enable inference features in intelligent systems.

In this context, we can point out different ontology models to represent knowledge domain. For instance, Web ontologies are used for distinct purposes and in several phases of Information Systems specification, design, and use. Soft ontology is another conceptual approach, in which, in contrast to Web ontologies, with fixed hierarchies

described in Web Ontology Language (OWL), refers to flexible set of meta-data [8]. This is useful to represent dynamically evolving information domains. These ontologies have individual elements associated with values in a non-structured *a priori* hierarchy.

Ontology Charts (OCs) in Organization Semiotics (OS) [10] play a central role in understanding the affordances, capture the behaviour of the involved agents, and the ontological dependencies at design time. OCs are the results from the Semantic Analysis Method (SAM), which aims to produce a clear ontological commitment concerning the nature of reality. In this sense, in SAM, ontology is understood in a metaphysical position [15]. This makes possible, with others OS' methods, a rigorous treatment of the Information Technology (IT) system at the non-technical semiotic levels [15].

Several types of ontological representations may coexist in computational systems. Thus, it is necessary to find related entities from one model into another to allow data exchange and cross-model interpretation. However, a key challenge is to enable the conciliation of data from computational systems (or part of the same system) that adopt these ontologies, because each system can interact with several knowledge domains in a variety of ways. We propose the concept of a network of ontologies to enable the coexistence of different models. The use of a network of ontologies opens up the possibility of mixing up various types of ontologies.

Although the literature has proposed several approaches to obtain interoperability among models and transformation rules (*e.g.*, [4]), we still need proposals suited to enable ontologies of distinct nature to coexist in the same conceptual space, *i.e.*, without thoroughly transforming one model into another. We assume that OCs and soft ontologies might coexist with Web ontologies that model more rigid hierarchies to represent domain concepts. Nevertheless, it is still an open research problem how to support the coexistence of these different models.

In this article, we propose a meta-model that allows the specification of models to support the connection of concepts of the distinct ontologies. Our proposal enables the use of a network of ontologies obtaining interoperability among the models. Given an entity in one model, our solution provides a way to identify connected entities into other models. The proposed meta-model defines the types of entities, their connections, and the beliefs associated to the connections (of any type). We describe the reached meta-model and formalize the main aspects involved.

Our approach is exemplified by assuming two computational systems (or subsystems) that must interoperate, systems *A* and *B*. In system *A*, the “*House*” class is described in an OWL model, and in system *B*, there is the affordance “*Home*” specified in an OC. The goal is not to assign that the “*House*” class is equivalent to the affordance “*Home*”. They are not equivalent (or same as) concepts, once “*House*” is defined as a class (from the OWL conceptual perspective) and “*Home*” is defined as an affordance (from OC perspective), as well as the terms are not semantically equivalent. Thus, our goal is to provide a meta-model that allows us to model how they are connected at the technical level allowing data exchange. The obtained results indicate the possibility of reconciliating ontology models of different nature via our proposal.

The remainder of this article is organized as follows: Section 2 presents the foundations and related work. Afterwards, we thoroughly describe the proposed meta-model (Section 3). Section 4 reports on an application scenario to exemplify our proposal. Section 5 discusses the findings and Section 6 wraps up the article.

2 Foundations and Related Work

In the Computer Science context, ontologies aim to represent semantics in computational systems [7] and have been designed to provide rich machine-decidable semantic representations. They refer to a formal specification of some domain, representing a conceptualization of a domain in terms of classes, properties and relationships between classes. Usually, a syntactic structure models the concepts of a knowledge domain and serves as schemes that organize data expressing instances of concepts according to logical properties. The classes are the focus of ontologies because they describe the domain concepts representing groups of individuals that share properties. Data properties characterize individual attributes, whereas object properties specify the relationships between individuals of the same or different classes.

Another relevant methodology to model a knowledge domain is the Semantic Analysis Method (SAM). SAM uses a subjective philosophical stance and an agent-in-action ontology to determine the underlying semantics of a social context and the relationship between the human agents and their patterns of behavior [13]. The SAM supports the analysis and representation of a social system. The method assists problem-owners in representing meanings in a formal and precise semantic model, the Ontology Chart (OC). This refers to a conceptual model that describes a view of responsible agents in the focal domain including their affordances, and the ontological dependencies between them [10].

Gibson [6] introduced the concept of Affordance to express the invariant repertoires of behaviour of an organism made available by some combined structure of the organism and its environment. In SAM, the concept introduced by Gibson was extended to include behaviour invariants in the social world [10]. Agent is a special kind of affordance, which takes responsibility both for their own actions and the others' actions, *e.g.*, an individual person, a cultural group, a society, *etc.*

Kaipainen *et al.* [8] proposed the concept of soft ontology to conceptualize dynamic domains. A soft ontology declares a set of meta-data to describe a domain by means of spatially conceptualized properties. According to the authors, soft ontologies can be used to represent evolving information domains, such as those of collaborative tagging. They explore the concept of *ontodimension*, as a finite set of items (terms), to define a domain. For each item, a degree (weight) is used to prioritize each ontological dimension. In a technical perspective, we propose to represent soft ontologies via RDF model. To accommodate the prioritization of terms, we use an adapted RDF fuzzy model [11].

Our proposal relies on the concept of metamodel as an alternative to define a “language” to design models. The aim is promoting the interoperability among

computational systems, which are based on heterogeneous ontological models. Considering a model as an artifact that provides abstraction of the reality via a description of concepts, a metamodel represents a model to describe what are these concepts, and how they are linked. Elements of the models refer to instances of the metamodel. By defining a metamodel, we address the challenge of identifying the elements of each model investigated in this study.

Regarding to previous studies on this matter, OS literature presents investigations aiming to promote the use of OCs with other models. Bonacin *et al.* [2] inquired the use of OC for system design at the technical level modelling with UML. The authors proposed a sequence of steps and a group of heuristic rules to construct UML class diagrams from OCs. In [1], the authors focused on the mapping of OCs to UMLs class diagrams preserving the semantic normal form. Other studies presented alternatives for the transformation from OCs into class diagrams [14].

Previous work also suggested a semiotic approach to design Web ontologies. Dos Reis *et al.* [3] proposed to enrich the representation aspects of OWL ontologies based on the use of Agents, Affordances and Ontological Dependencies to describe OWL classes. For this purpose, existing study defined heuristics for semi-automatic generation of Web ontologies from OCs [5]. The heuristics created explicit relations between the elements of the models to construct Web ontologies from OCs. A computer assisted process was defined to build the Web ontologies [4] and implement such heuristics. The authors defined transformation rules for deriving an initial Web Ontology described in OWL from OCs.

In the context of Web ontology integration, Umer and Mundy [17] investigated an ontology integration process relying on three different scenarios: ontology reuse, ontology mapping and ontology merging. These authors proposed a framework for ontology integration composed of three layers (Syntax Analyzer, Semantic Analyzer and Taxonomy Analyzer). Their approach compared ontology's concepts based on their syntax, semantics and relationship between concepts. Similarly, Nguyen and Truong [12] investigated the integration of fuzzy ontologies addressing critical issues related to different fuzzy structures of concepts and relations. The authors proposed a two-step approach relying on a distance calculation of the involved datasets and applied a consensus algorithm to obtain the integration.

In the context of authoring tools, Timm *et al.* [16] analyzed the usefulness of model-driven development to speed up the process of building semantic web services. Their work associated OWL-S specifications via the development of standard UML models. The authors argued that by using transformations from equivalent UML constructs, difficulties caused by a steep learning curve for OWL-S can be mitigated with a language that has a wide user base.

Literature has presented key aspects to be considered in ontology integration and transformation between distinct models. However, mostly of the existing studies focus on the integration of ontologies described in an equal language, or they have emphasized those models that share similar philosophical stances. In this investigation, unlike the others, we define a metamodel to facilitate the development of ontology networks, which accommodate models with different conceptions

(including various language) for promoting data exchange and cross-ontology interpretation.

3 Constructing the Metamodel

The first step was the definition of the goals of the metamodel, which can be summarized as follows: (1) it should allow to represent connections between entities of various models, aiming to promote data exchange and cross interpretation in networks of ontologies; (2) it should assume a neutral position, as much as possible, with few assumptions and premises; and (3) it should be restricted to modeling the connection between models, *i.e.*, it should not emphasize the support of the modelling of internal aspects of ontologies. In the second step, we identified the building blocks that compose the target models we aim to conciliate. These elements enabled us to understand the connected entities along the models. The final step was the definition and formalization of the metamodel based on the detected correspondences.

We analyzed a set of modelling approaches, including OWL, OC, fuzzy soft ontology, Simple Knowledge Organization Systems (SKOS) and RDF(S). The first three were selected for the development of a proof of concept. These models represent different approaches for semantic modelling, whose key characteristics can be briefly summarized as follows: (1) OWL is a description logic-oriented language, with formal and rigid descriptions allowing logical inferences and computer interpretation; (2) OC focuses on the agents, their patterns of behaviors, and ontological dependencies, where ontology is understood in a metaphysical position; and, (3) Fuzzy soft ontologies emphasize less rigid representations enabling dynamic relationships and represent aspects related to uncertainty.

Based on the above assumptions, we defined M_2 as our interoperability metamodel that represents concepts to interrelate the different models. We developed an instance of the metamodel defined as the M_1 , which aims to interrelate the three targets model. The M_0 stands for the instances of the M_1 models. Section 4 presents an application scenario with examples of M_0 instances.

Figure 1 illustrates our metamodel (M_2) in a graphical representation based on conceptual maps. The concept “*Language*” is used to describe Entities from a model. For instance, the OWL language describes Classes and Object properties, whereas the OCs language describes Affordances and Agents. The concept “*Connection*” is central in the proposal. This concept enables creating relationships from an entity in a model *A* to a model *B*. The “*Connection*” has a description via the element “*Connection_Description*”.

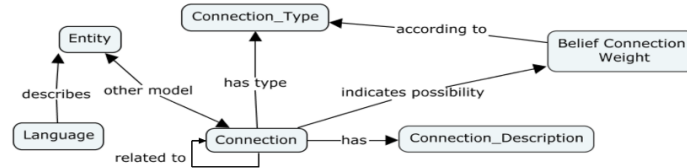


Fig. 1. Representation of the proposed metamodel M_2

We represent the nature (e.g., conjunctive, disjunctive or assertive) of the “Connection” via the element “Connection_Type”. The metamodel expresses that a “Connection” has a “Connection_Type” with a degree of belief. To describe it, the element “Belief Connection Weight” associates a value ranging $[0,1]$ according to the “Connection_Type”; alternatively, a discrete scale can be used to represent degrees of belief. The M_2 enables the “Connection” with various instances of entities and self-connections. This allows representing that a “Connection” c_1 can be related to another different “Connection” c_2 .

Figure 2 presents the M_1 as an instance of the M_2 . We assume, in a simplified view, that the OWL language describes the following set of Entities: {*Classe*, *Object Property*, *Data Property*, *Instances*}. Considering the OC as instance of “Language”, it describes the following Entities: {*Affordance*, *Agent*, *Role*, *Ontological Dependence*, *Determiner*}. Similarly, we assume the entities for the soft ontology based on a Fuzzy RDF language as follows: {*Subject*, *Predicate*, *Object*, *Type*, *Val Deg*}.

Figure 2 shows the instances of Languages that describe the Entities standing for all possible elements that the languages express. We consider that a “Connection” is expressed by a unique identifier *URI*. The concept “Connection_Type” can assume three values as *conjunction*, *disjunction*, and *assertion*. This enables to relate one connection with another. As a *disjunction*, the model expresses that if a connection is of one entity instance, it cannot be of another entity. The “Belief Connection Weight” describes a degree of confidence between the “Connection” and its type. It expresses a fuzzy value or even a qualitative value. The “Connection_Description” enables us to describe the relationship that a “Connection” determines. This describes, for instance, that an entity in one model is an “instance” of another model, or it is “part_of”.

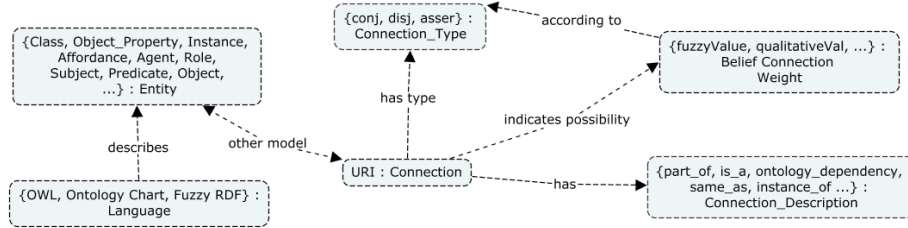


Fig. 2. Example of Instance of the Metamodel (M_1)

Figure 3 presents an excerpt of formalization of our metamodel using an extended *Backus-Naur* form (eBNF) based notation, as used in the OWL specification. Simplicity was a premise in this specification, which can be extended to support complex logical descriptions, such as data transformation and axioms.

```

OIDM ::= 'Oidm' ( ' [ oidmID ] { language } ' )
language ::= 'Language' ( ' languageID langName { axiom } ' )
axiom ::= entity | connection
entity ::= 'Entity' ( ' entityID entityName ' )
connection ::= 'Connection' ( ' connectionID [connectionName] connectionRestriction
connectionDescription ' )
connectionRestriction ::= connectionType ' ( ' connectionRestrictionID entityID
{entityID} beliefConnectionWeight ' )
connectionType ::= 'DisjointConnection' | 'ConjunctiveConnection' | 'AssertCon-
nection'
connectionDescription ::= 'is a' | 'part of' | 'same as' | 'instance of' | plainLiteral
beliefConnectionWeight ::= floatingPointNumber | valueScale
floatingPointNumber ::= bindigit [ '.' digit {digit} ]
digit ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
bindigit ::= 0 | 1
valueScale ::= ' [ ' {plainLiteral ' , ' } plainLiteral ' ] '
langName ::= plainLiteral
entityName ::= plainLiteral
connectionName ::= plainLiteral
connectionTypeName ::= plainLiteral
oidmID ::= URIreference
languageID ::= URIreference
entityID ::= URIreference
connectionID ::= URIreference
connectionTypeID ::= URIreference
connectionRestrictionID ::= URIreference
/* reused from OWL */
plainLiteral ::= lexicalForm — lexicalForm@languageTag
lexicalForm ::= as in RDF, a unicode string in normal form C
languageTag ::= as in RDF, an XML language tag

```

Fig. 3. Excerpt of the metamodel eBNF Code

4 Application Scenario

We present three examples of M_0 related to the same application scenario to illustrate the use of our metamodel to integrate OWL ontologies, Ontology Chart and soft ontologies. Figure 4 includes an example of disjunction using our metamodel. This scenario connects “John” (in M_0), as instance of Subject in Fuzzy RDF (M_1). The subject “John” can be connected to two disjoint possibilities. The model expresses that there is a belief “John” is instance of a *Dog* Class in an OWL ontology, or “John” is connected to a *Person* Affordance in an OC. The “Belief Connection Weight” is associated with the pair “Connection_Type” (disjunction) and the respective connection (0.9 for *Person* and 0.1 for *Dog*).

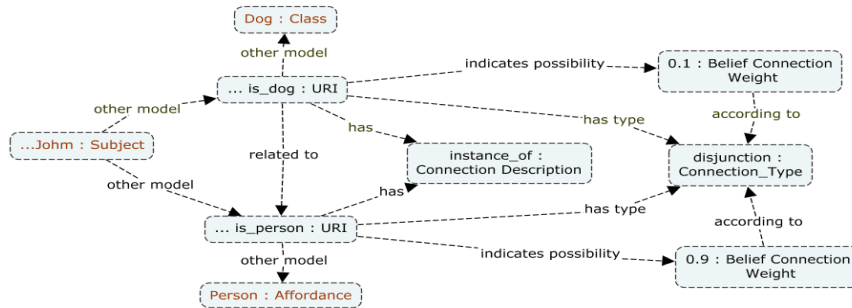


Fig. 4. Example: Connection type *DisjointConnection*

Figure 5 presents an example of conjunction. In this scenario, as defined by the conjunction *Connection_Type*, “John” can be either employed in *CTI* and *FACCAMP*. In this case, the “*Belief Connection Weight*” determines the degree to which we believe this type of connection holds for them (0.9 for *CTI* and 0.7 for *FACCAMP*). Note that both connections are related to each other.

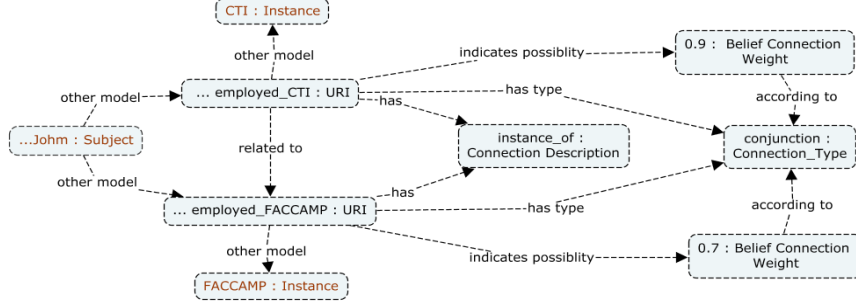


Fig. 5. Example: Connection type *ConjunctiveConnection*

Figure 6 presents an example to express a statement regarding the *sameAs* assertion. In this case, a RDF fuzzy model defines the degree to which “John” is *Anger* and *Sadness* (this is an internal representation of the RDF fuzzy model). Our metamodel enables to relate that the *Anger* object is equivalent to an OWL Class in another model with 95% of belief. Similarly, we relate the object *Sadness* to an Affordance *Sadness* in an OC.

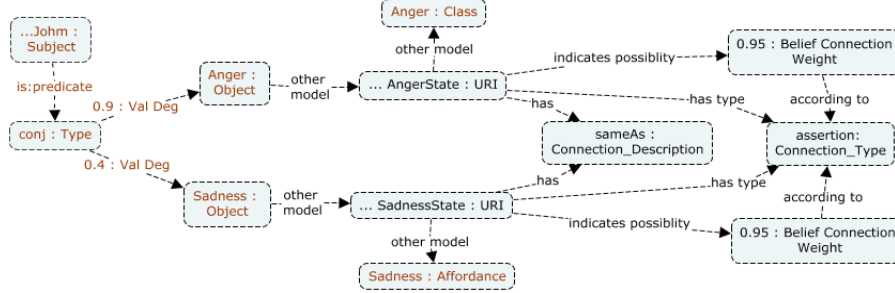


Fig. 6. Example: Connection type *AssertConnection*

5 Discussion

The formal nature of ontologies makes them amenable to machine readability and provides a well-defined semantics for defined terms. Nevertheless, the use of different formalism to represent a knowledge domain prevents the properly interoperability between different systems. This is not even mitigated when considering the use of adopted formalisms such as Web ontologies, ontology charts and soft ontologies.

This work dealt with the conciliating of different ontology formalisms by proposing a metamodel that allows the construction of a network of ontologies. Our formal metamodel defined elementary building blocks that are instantiated to correspondent elements in the target formalisms.

We found that is possible to accommodate conceptual entities in the metamodel to create data connections from one model to another, *i.e.*, without transformation operations. We explored the concept of Belief Weight, inspired by RDF Fuzzy models, to determine the degree of confidence between elements from different model formalisms. This aspect refers to the key originality in this research. The presented application scenario showed the benefits and limitations of our findings. The practical examples illustrated how a concept in a Web ontology was connected to an Affordance in OCs with a given degree of confidence based on our model. The examples explored the different types of connections to illustrate the effectiveness of the proposal.

Our metamodel is still unable to represent formal axioms from Web ontologies as well as OS' norms, however our model focuses on supporting data interoperability. Future work will involve a refinement of the metamodel in addition to implement and test the provided formalization. We plan to study how our proposal can be employed in scenarios where ontologies evolve over time, since new concepts are created, modified, and removed along the system execution.

Another challenge to be undertaken is how to make inferences in “cross models” representations. This is a known difficulty in distributed ontology networks and models. This difficulty is furthermore aggravated by the fact of dealing with heterogeneous networks. Practical issues, such as the performance using real-world ontologies and systems should also be analyzed.

6 Conclusion

Ontologies are central artifacts to represent meanings in information systems. However, different conceptual models are hardly conciliated in the same conceptual space. In this paper, we proposed a metamodel to enable different ontology models to coexist in a network of ontologies. Findings revealed that the metamodel is suited to represent distinct connection types considering a belief connection weight. The presented examples illustrated the findings in an application scenario. Further aspects such as inference strategies, deepening in theoretical issues, and the use of the metamodel in real-world situations shall be investigated. Our future work mostly involves addressing the dynamic aspect of the ontology network, the study of the inference issues, and the application of the metamodel in the specification of a network for an enactive system.

Acknowledgements

This work is supported by the São Paulo Research Foundation (FAPESP) (Grants #2015/16528-0 and #2017/02325-5).

References

1. Ades, Y., Umar, F.B., Poernomo, I., Tsaramirsis, G.: Mapping ontology charts to uml: an snf preserving transformation. In: Proceedings of the 10th International Conference on Organisational Semiotics (2007).
2. Bonacin, R., Baranauskas, M.C.C., Liu, K.: From ontology charts to class diagrams: Semantic analysis aiding systems design. In: 6th International Conference on Enterprise Informaiton Systems (ICEIS'04). pp. 389-395 (2004).
3. Dos Reis, J.C., Bonacin, R., Baranauskas, M.C.C.: A semiotic-based approach to the design of web ontologies. In: 12th International Conference on Informatics and Semiotics in Organisations (ICISO'10). pp. 60-67 (2010).
4. Dos Reis, J.C., Bonacin, R., Baranauskas, M.C.C.: An assisted process for building semiotic web ontology. In: 13th International Conference on Informatics and Semiotics in Organisations (ICISO'11). pp. 167-174 (2011).
5. Dos Reis, J.C., Bonacin, R., Baranauskas, M.C.C.: Constructing web ontologies informed by semantic analysis method. In: 13th International Conference on Enterprise Information Systems (ICEIS'11). pp. 203-206 (2011).
6. Gibson, J.J.: "The theory of affordances," in *Perceiving, Acting, and Knowing. Towards an Ecological Psychology*. Hoboken, NJ: John Wiley & Sons Inc. (1977).
7. Gruber, T.R.: Toward principles for the design of ontologies used for knowledge sharing. *International Journal of Human Computer Studies* 43(5-6), 907-928 (1995).
8. Kaipainen, M., Normak, P., Niglas, K., Kippar, J., Laanpere, M.: Soft ontologies, spatial representations and multi-perspective explorability. *Expert Systems* 25(5), 474-483 (2008).
9. Kalyanpur, A.; Golbeck, J.B.J.H.J.: Owl: Capturing semantic information using a standardized web ontology language. *Multilingual Computing & Technology Magazine* 15(7) (2004).
10. Liu, K., Li, W.: *Organisational Semiotics for Business Informatics*. Routledge, Abingdon (2014).
11. Lv, Y., Ma, Z.M., Yan, L.: Fuzzy rdf: A data model to represent fuzzy metadata. In: IEEE International Conference on Fuzzy Systems (IEEE World Congress on Computational Intelligence). pp. 1439-1445 (2008).
12. Nguyen, N.T., Truong, H.B.: A consensus-based method for fuzzy ontology integration. In: Pan, J.S., Chen, S.M., Nguyen, N.T. (eds.) *Computational Collective Intelligence. Technologies and Applications*. pp. 480-489. Springer (2010).
13. Salter, A; Liu, K.: Using semantic analysis and norm analysis to model organisations. In: Proceedings of the 4th ICEIS. pp. 847-850 (2002).
14. Santos, T.M.D., Bonacin, R., Baranauskas, M.C.C., Rodrigues, M.A.: A model driven architecture tool based on semantic analysis method. In: Proceedings of the ICEIS. pp. 305-310 (2008).
15. Stamper, R., Liu, K., Sun, L., Tan, S., Shah, H., Sharp, B., Dong, D.: Semiotic methods for enterprise design and it applications. In: Proceedings of the 7th International Workshop on Organisational Semiotics (OS) (2004).
16. Timm, J.T.E., Gannod, G.C.: A model-driven approach for specifying semantic web services. In: *International Conference on Web Services*. pp. 313-320 (2005).
17. Umer, Q., Mundy, D.: Semantically intelligent semi-automated ontology integration. In: *Proceedings of the World Congress on Engineering* (2012).