

Towards supporting business services discovery through the integration of organizational models with ontologies

Blanca Vazquez^{1,2}, Alicia Martinez², Anna Perini³, Hugo Estrada^{1,2}
and Mirko Morandini³

¹ Fund Information and Documentation for the industry, Av. San Fernando 37, Col. Toriello Guerra, Tlalpan, 14050 Mexico, D.F., Mexico

{blanca.vazquez, hugo.estrada}@infotec.com.mx

² National Center of Research and Technological Development, Interior Internado Palmira S/N, 62490 Cuernavaca, Morelos Mexico

{blancavazquez11c, amartinez, hestrada}@cenidet.edu.mx

³ Bruno Kessler Foundation - IRST, Center for Information Technology, Via Sommarive, 18, 38050 Trento, Italy

{perini, morandini}@fbk.eu

Abstract. We argue that the integration of organizational models with ontologies is useful to discover business services because we consider that a domain concept that integrated different elements of the model could represent new business services to the organization. These additional services could improve the performance of an enterprise and detect new opportunity areas. Moreover, a model annotated with domain concepts improves the analysis process because included additional information can facilitate a semi-automated reasoning among elements. This paper presents an approach to integrate visual models of an organization with a general or specific domain ontology and proposes semantic annotation suggestions. A software tool is presented to implement the integration process, and an illustrative example validates the effectiveness of our approach.

Keywords: visual models, ontologies, OntoSem, integration process, business services

1 Introduction

In recent years, considerable attention has been paid to organizational modelling techniques. This interest is motivated by the need of achieving a better understanding of organizational knowledge, and to support the definition and implementation of suitable business processes. The use of a visual model to represent the organizational behavior allows the analyst to identify relationships among stakeholders and their mutual dependencies. From the perspective of the business processes in an organization, visual models help to make explicit the

strategic goals behind them, the social context, the resources and their structure. From a business process engineering perspective, ideally, we would like to be able to derive the design and the discovering of business services from an organizational model. However, there are still some gaps and weaknesses in the global engineering process, for instance in discovery of business services, in the integration between organizational models with ontologies and in the addition of semantic information to existing models.

The organizational model helps to detect the relationships between stakeholders, to describe the tasks, resources and softgoals that are needed to achieve a common goal, and to detect bottle-necks. We consider that organizational models, integrated with a domain ontology, could help to discover new business services which emerge from existing services.

The main contribution of this paper is to propose an approach for integrating organizational visual models with general and specific domain ontologies. We claim that the integration of an organizational model with an ontology can improve the performance in discovering business services, because we consider that a concept that integrates different model elements could represent new business services to the organization, and this additional services could improve the performance of an enterprise detecting opportunity areas and improvements to the business. The advantages of such an integration would be to improve the process of analysis, search and reuse of information. The proposed approach is based on visual models represented in the i^* framework [1], in Tropos [2] and in service-oriented i^* [3]. The integration process is supported by the *TAGOO_{n+}* tool.

This paper is structured as follows: Section 2 gives a brief background of the relevant concepts used in our approach, Section 3 describes our approach to integrate organizational models with ontologies and describes the architecture of the proposed tool. Section 4 describes an illustrative example to provide a preliminary evidence of its feasibility and performance. Section 5 presents the lessons learned, Section 6 the related works and Section 7 the conclusions.

2 Background

In this section, we present a brief description of the relevant concepts of our approach: organizational models, ontologies and also presents an introduction of OntoSem the which is a general ontology.

2.1 Organizational models

An organizational model describes the business behavior of an organization representing graphically the stakeholders and the relationships among them. The development of organizational models has been studied in the fields of requirements engineering and organizational process modelling [4]. We propose to apply our approach to organizational models described in the i^* variants *Tropos* and *Service-oriented i^** . The i^* framework [1] is a goal-oriented and agent-oriented

modelling framework. i^* allows modelling the actor’s goals and dependencies among them, and it includes a graphical notation and two models to represent the organizational knowledge. Tropos [2] is a methodology that adopts i^* primitives to model business from early requirements to software agents implementation. Service-oriented i^* [3] is an extension of i^* to model an enterprise from a service-oriented perspective. i^* , Tropos and Service-oriented have been applied in different fields [5–7].

2.2 Ontologies

An ontology is an explicit specification, a formal and shared conceptualization [8], based on the idea of a simplified conceptualization of the world. Ontologies are useful e.g. to facilitate the integration of information defined across individual languages. Their semantics are thus specified in an unambiguous way. According to [9], ontologies are classified in (i) *upper level ontologies* or *general ontologies* that describe general concepts like space, time, matter, object, event, action, etc. These ontologies are independent of a particular problem or domain; (ii) *domain ontologies* that describe the vocabulary related to a generic domain (like medicine or automobiles). Each concept in such an ontology is called “*domain concept*”; (iii) *application ontologies* that describe concepts depending both on a particular domain and task.

2.3 The OntoSem ontology

The OntoSem (*Ontological Semantics*) [10] ontology is a formal, language independent, unambiguous general ontology that provides a metalanguage for describing conceptual meaning. The root concepts of OntoSem are *object*, *event* and *property* (Fig. 1). It contains around 8000 concepts and 350 properties with their respective meanings. Various other general, foundational ontologies are available, such as [11, 12]. A relevant feature of OntoSem compared with these ontologies is to be a practical ontology. OntoSem has been practically validated in several successful applications [13, 14].

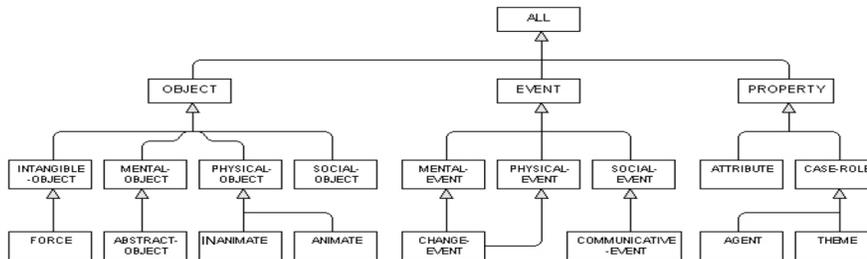


Fig. 1. Fragment of the OntoSem ontology showing its root concepts: object, event and property.

3 Integrating visual models with ontologies

We propose the integration of organizational models represented in the i^* variants Tropos and Service-oriented i^* , with a general or a specific domain ontology. In order to carry out the integration of a visual model with a domain ontology, we define two phases: *i) Semantically annotating the organizational model* and *ii) Integrating the organizational model with the ontology*. The first phase consists of annotating the organizational model from of generals and specifics semantic annotation suggestions, these suggestions are the guideline to annotate the model elements, the annotation process is performed by the final user (analysts). The second phase consists of integrating the previous annotated model represented in iStarML format with general or specific domain ontology. In Figure 2 the two phases of the proposed approach are shown.

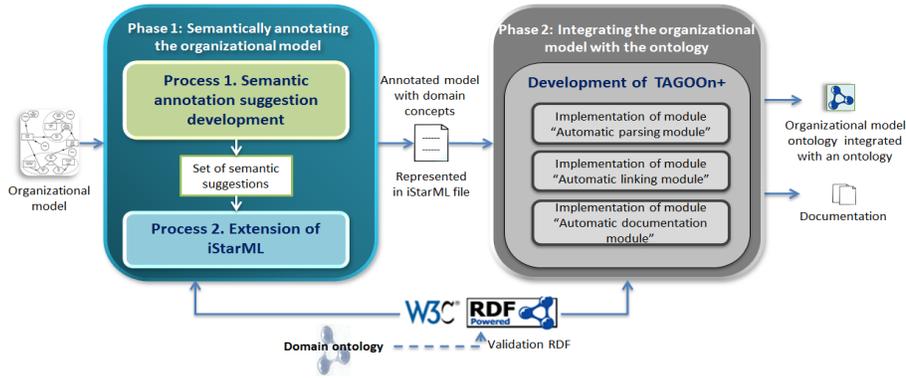


Fig. 2. Approach to integrate a visual model representing in i^* , Tropos or Service-oriented i^* with a general or specific domain ontology.

3.1 Phase 1: Semantically annotating the organizational model

A set of *general* and *specific* semantic annotation suggestions support this phase. The *general suggestions* should be applied to domain ontologies in general. The *specific suggestions* are domain-dependent and instantiated in the *OntoSem* ontology.

The process to develop the set of semantic annotation suggestions is described in [15, 16]. The suggestions are the key to annotate the organizational models. The annotation is carried out by the analyst first following these semantic annotation suggestions. Then, the analyst should go in-deep in the selected ontology to find out the most appropriate domain concept for each model element. The concept selected to annotate an element should be congruent with its existing description. To be able to share such annotated models, we propose to represent them in an interchange format such as *iStarML* [17]. *iStarML* is an XML-based

proposal for i* models interchange, built taking in consideration several meta models of i* variants. We propose an extension of iStarML by adding an XML attribute *sannotation*[15, 16]. This attribute allows us to identify each element of the model with its respective semantic annotation. We use jUCMNav for organizational modelling. In order to automate the creation of an iStarML file from an annotated model, we extended an existing jUCMNav plug-in. The analyst should use the extended plug-in for the generation of an annotated model represented in iStarML.

3.2 Phase 2: Integrating the organizational model with the ontology

The second phase consists of integrating the annotated organizational model with the ontology used for the semantic annotation. We present a tool called *TAGOOn+* (*Tool for the Automatic Generation of Organizational Model Ontologies and Integration*) to carry out this integration. TAGOOn+ is an extension of the TAGOOn tool [18]. The original tool transforms i* based models into ontologies, including the i* variants, Tropos and Service-oriented i*. Our extension consist of integrating the transformed model in ontologies with a general or specific domain ontology. The documentation of this integration is generated automatically by our extended tool. We consider that this documentation can be used by the analyst for correctly comprehending the concepts used in the model.

The inputs of TAGOOn+ are: i) the annotated model represented in an iStarML file, and ii) the ontology represented in an OWL file. The result is an organizational model integrated with a general or specific domain ontology, and the documentation of this integration represented in a text document. In order to carry out the proposed integration, the annotated model needs to be transformed into an organizational ontology. The meta-ontology *OntoiStar+* [18, 19] promotes the transformation from an organizational model to an organizational ontology. *OntoiStar+* provides a joined ontological representation of the meta-models of i*, Tropos and Service-oriented i*. We extend this meta-ontology to capture annotated models, by adding the data property “*Node_sannotation*”, which stores the semantic annotation of each instance of the model.

The integration of an organizational model (represented in ontological concepts as shown in the reminder) with the ontology used for its annotation, results in an ontology witch captures the relationships between goals in a goal model *and* the relationships between the concepts described in the goals (through their semantic annotation) captured in the domain ontology.

The most important feature of general ontologies is their hierarchical structure, where all the concepts are grouping according similar meanings facilitating the reasoning among concepts. For instances in *OntoSem*, the concept *bank* presents a relationship of type is “is_a” with the concept *Financial Corporation*, and this concept also presents a relationship of type is “is_a” with the concept *Private-organization*. Supposing, that a model element of type actor called “*International Bank*” was annotated with the concept *Bank*, so the analyst can infer that the *International Bank* is a *Financial corporation* and also it is a *Private organization*. For instance, the concept *Bank* means *A financial institution that*

accepts deposits of money from and loans money to the public. In the previous example, the analyst infers that a bank is a Financial Corporation and is an institution that manage money and accepts deposit and loans of the public.

In this way, we consider that a visual model integrated with ontologies can improve the process of analysis in the organizational models and to achieve the semi-automated reasoning between the elements supported by the structured concepts in the ontology. Moreover, each concept in OntoSem provides a particular description. When an organization is modeled by an analyst using some i^* variant is not possible to add all this information in a single label. In this way, when a model element is annotated with one or more domain concepts, the information provided by a general ontology is useful to clarify the description of the each model element avoiding ambiguities, facilitating the understanding inside the organization, searching and reuse of information.

3.3 Architecture of TAGOOn+

TAGOOn+ supported the automatic transformation and integrating of an i^* based model represented in the variants: i^* , Tropos and Service-oriented i^* with a general or domain ontology. TAGOOn+ is the tool used by the analyst to integrate an organizational model with a specific ontology. The architecture of TAGOOn+ tool (illustrated in Fig. 3) is based on three modules: “*Automatic Parsing Module*”, “*Automatic Linking Module*” and “*Automatic Documentation Module*” (Fig. 2), built around the TAGOOn “*Automatic Mapping Module*”, in order to guide the integration process as detailed below:

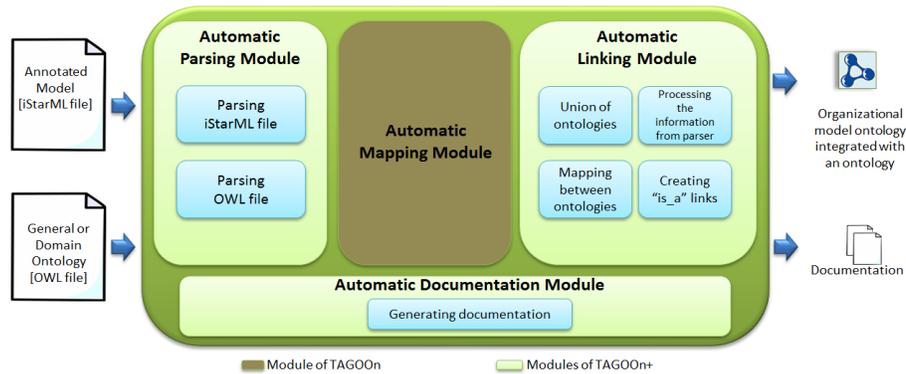


Fig. 3. Architecture of the TAGOOn+ tool

Automatic Parsing Module. The first module consists of two submodules. The first submodule *Parsing iStarML file* consists of reading the annotated model represented in an iStarML file, parsing the fields: *id*, *name*, *type*, and *semantic annotation* of each instance of the model, and its relationships with

other elements. An array with the information of each field is generated in this submodule. The second submodule *Parsing OWL file* consists of reading the ontology represented in an OWL file. The result is an array that stored the names of domain concepts and the description of each concept.

Automatic Mapping Module. The second module performs the automatic transformation from an i* based model into an ontology derived from the concepts of OntoiStar+. In an ontology a *data property* link resources to literal values, the labels of *id*, *name* and *type* of each instance of the model are represented as data properties in the generated ontology. The semantic annotation is stored in a data property called “*Node_sannotation*”.

Automatic Linking Module. The third module consists of four submodules. The first submodule *Union of ontologies* consists of integrating the organizational model represented as ontology with a general or specific domain ontology. The concepts of both ontologies are integrated in an OWL file. The second module *Processing the information from parser* consists of reading the arrays obtained in the first module. First, we obtain the name of each element and its semantic annotation, after the name of domain concepts are extracted. Each term is converted from upper case to lower case, and the elimination of white spaces, slash and other information not necessary are carried out. This process is necessary to avoid wrongs during the integration process.

The third submodule *Mapping between ontologies* consists of comparing each data property “*Node_sannotation*” of the model elements with the names of concepts. If both values are equal, then is saved the URI of the concept and the model element. For instance, a task element labeled as *Capturing student data* is annotated with the domain concept *record-information*. The submodule of mapping compared this concept with all the domain concepts on the ontology, and if the concept *record-information* is found, then the URI of task element (domain) and the domain concept (range) will be saved. The semantic annotation saved in the iStarML file helps only to assurance that each model element is integrated with one or more domain concept of the general or domain ontology.

The fourth submodule *Creating “is_a” links* it is the most important submodule of all the architecture of TAGOOn+, due to related each model element with one or more domain concepts of the selected ontology. The URI of the domain and range of the previous submodule is used to establish the relationship between the model element and the concept of the ontology. If a model element does not contains semantic annotation then it will be not related with any concept of the ontology. The result of this submodule is the OWL file integrating both ontologies, where each model element should relate with one or more domain concepts. We propose to use Protégé to visualize the generated ontology by TAGOOn+.

Automatic Documentation Module. The fourth module *Generating documentation* consists of reading the arrays generated in the first module. If the ontology provides a description of each concept, by a metalanguage or by conceptual relationships, we generate a document that describe the name of each model element together with its semantic annotation and the description read out of the ontology. We consider that the documentation generated by TAGOOn+ is useful

for the technical and analyst people in order to achieve a better understanding of the organizational model describing each model element with its meanings.

4 Illustrative example

In order to validate our approach, we applied it to several examples. In this paper, we start from an i* strategy dependency model that describes a smart-card based payment system which has been previously modelled in [20]. The process to integrate an organizational model consists of two steps. The first step consists in annotating an visual model applying the semantic suggestions, and to represent the model in the iStarML format. As second step, “*Integrating ontologies*” consists of representing the visual model in an ontology and to integrate it with the general ontology used for its annotation. The details are presented below:

Phase 1: Annotation. We annotate the visual model following a set of specific semantic annotation suggestions presented in [15]. An example of specific suggestions for the goal element is: $ME : Goal \xrightarrow{AB} SC : mental - event \wedge SC : social - event \wedge SC : mental - object$; where “*ME*” means *Model Element*, “ \xrightarrow{AB} ” means *can be annotated* and “*SC*” means *SuperConcept*. In this illustrative example, we used the specific suggestions for OntoSem described in [15]. For instance, the illustrative example presents a goal element *Present Card for Transaction*. We need to find appropriate domain concept to describe this element to provide a precise, formal meaning to the element, thus making it more understandable to people and for automated techniques.

According to the previous specific suggestion, a goal element can be annotated with the super concepts *mental-event*, *social-event* and *mental-object*. Then, going in-depth to the OntoSem ontology we search for more detailed domain concept for each instance of the model. For our goal element *Present Card for Transaction*, the concept *identify* from *mental-event* describes “*to fix the identity of something or someone*”, moreover the concept *authenticate* from *social-event* describes “*to verify the identity of someone or something in order to grant access privileges*”, finally the concept *negotiate-transaction* from *social-event* describes the goal “*to work out the terms of a transaction in order to reach an agreement*”.

In this way, the goal element *Present Card for Transaction* is annotated with the concepts *identify*, *authenticate* and *negotiate-transaction*. Each domain concept selected adds information to the goal element, in this way the additional information is related with verifying the identity in order to grant access privileges or to reach an agreement, in this case, to start a transaction, this information is not visible with a single label, but when semantic annotation is added to the model the description is improved to avoiding ambiguity and to share a same knowledge. In Fig. 4 the hierarchical view of these concepts is shown. A fragment of the annotated strategic dependency model is shown in Fig. 5. The analyst (or the supporting tool in our case) adds the semantic annotation in the model using the demarking symbol “@”.

It is important to point out that the domain concepts selected for each model element need to be congruent with the description of the model element, and also that the annotation process is manual. After annotating all the model elements, the model is exported to iStarML format. In Fig. 5 on the bottom side, the representation of goal elements in iStarML format with its semantic annotation is shown.



Fig. 4. Hierarchical of domain concepts “identify” (left), “authenticate” (center) and “negotiate-transaction” (right) to annotate the goal element “Present Card for Transaction”

Process 2: Integrating ontologies. In order to integrate the annotated model represented in iStarML format with the OntoSem ontology, first it is necessary to transform the model into an organizational ontology using TAGOOn+ and then to integrate it with the general ontology. In TAGOOn+ the strategic dependency model represented in iStarML and the OntoSem ontology represented as OWL are parsed and the information of each label are saved in two arrays. Then, in the tool the annotated model is transformed to an organizational ontology and the semantic annotation of each model element is saved in the data property “*Node_sannotation*”. Both ontologies are integrated to an OWL file and the information of each array is processed to avoid inconsistency during the mapping. The value of each annotation is compared with the domain concepts and matching concepts are saved for further processing.

Then, the tool creates relationships of type “is_a” are created between individuals of the model and the relative concepts of the general ontology. Creating relationships of inheritance, a model element is a subtype of a domain concept.

Finally, the information saved during processing helps to generate the documentation of the integration. Each model element related with one or more elements should describe its meanings. This information allows us to clarify and add additional information to the label of the model elements. In Fig. 6 the goal *Present Card for Transaction* is shown. This element was annotated with the domain concepts “identify”, “authenticate” and “negotiate-transaction” taken from the OntoSem ontology. The representation of this element using the extension of iStarML is also presented.

On the center side of Fig. 6, a fragment of model integrated with general ontology is shown. On the left table the goal element *Present Card for Trans-*

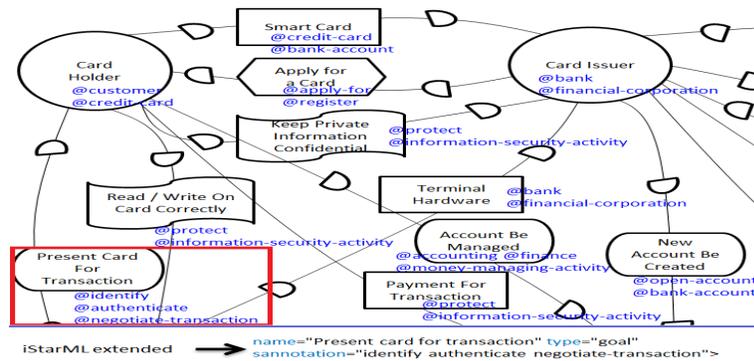


Fig. 5. Fragment of annotated strategic dependency model

action has relationships with the domain concepts *authenticate*, *identify* and *negotiate-transaction*, and their data property “*Node_sannotation*” presents the same information. On the right side, this information is presented graphically. This view permits to analyse with detail each model element. In this example it is possible to infer that *Present Card for Transaction* is referred to a financial-event, and also is an information-security-activity to start a transaction, and an analytic-cognitive-event. In this way, a model element annotated with domain concepts taken from an ontology allows us to infer new knowledge missing when it have only a single label. Moreover, the analyst could improve its process since the additional information helps avoiding ambiguity, facilitating the reuse of information of the model elements when creating new models, and also enables the detection of cross-item relationships.

The discovery of new services inside the organization is achieved when one or more domain concepts are used within different model elements. For instances, the illustrative example (Fig. 5) contains a softgoal dependency *Keep Private Information Confidential*, a resource dependency *Payment For Transaction* and a softgoal dependency *Read/Write-On Card Correctly*, that are annotated with the domain concepts *Protect* and *Information-Security-Activity*. We envision that these domain concepts could represent a more general service *Protect Information*, like “*PayPal*”. *Paypal* could represent a new business service, which allows a registered customer to use *PayPal* to pay its transactions, and its data are kept protected and privated thanks to this on-line payment processing service. A concept that integrated different model elements could improve the analysis process of the organization, because it is most feasible to infer new knowledge grouping different model elements than element by element, without any additional reference. Moreover, the additional information given by domain concepts makes the model more understandable to people and allows a further analysis. The documentation generated by TAGOOn+ describes the name, concept and its description for each element (see bottom side in Fig. 6).

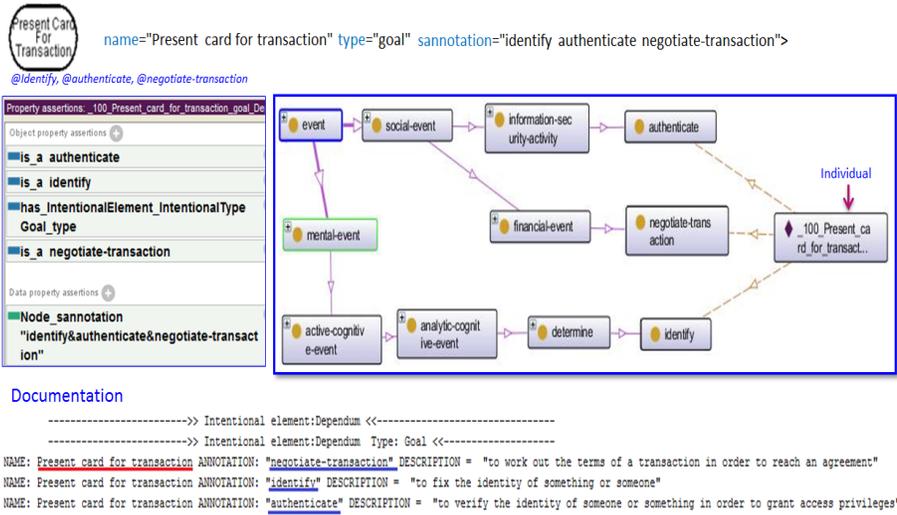


Fig. 6. Strategic dependency model integrated with the OntoSem ontology

5 Lessons learned

This approach addresses the problem of weaknesses of current business modelling techniques to define new business functionalities taking as base an organizational visual model. We consider that when a domain concept (taken from an ontology) is integrated in different model elements (tasks, goals, resources, soft-goals, etc.) this domain concept could represent a new functionality inside the organization. Often, the organizational model does not offer a clear information, moreover there can be redundancy of information, resulting in models not clear for the people who did not take part in the construction.

We believe that the hierarchical structure of ontologies is an advantage because it allows us to integrate each model element according to similar descriptions or situations facilitating a semi-automated reasoning among the model elements. The results of this and other illustrative examples lead us the following observations:

i) Using an ontology formed by more than 8000 concepts, the strategic dependency model was annotated using 49 concepts. We consider the integration process optimal due to the model transformed to an ontology, and then integrated with a domain ontology. The integration can thus be carried out without redundancy and inconsistencies.

ii) The integration of an organizational model with an ontology groups the model elements according to similar descriptions, exploring the relationships in the ontology, to achieve a better understanding of the organizational model. An annotated element offers more information to the organization because the hierarchy of associated domain concepts is available. For instance, the analyst can distinguish if an element is an object or an event, if the element is an institu-

tion, people or a service. In this way, the domain concepts eliminate ambiguity in labeling and thus facilitate to share knowledge.

iii) The proposed integration of an organizational model with ontologies through domain concepts permits the standardization of model elements according to similar situations or descriptions, allowing the analyst to reuse parts when creating new models, to detect additional cross-item relationships and to apply OWL reasoning techniques for completeness and consistency checking.

6 Related works

The integration of organizational models with a general or domain ontology in order to discover business services from the existent services has been poorly explored in requirement engineering. However, several approaches apply general ontologies focused on the issue of ontological interpretation for organizational modeling elements. For instance, in [21] a set of real-world semantics is applied to the modeling primitives of the ARIS Method (ARchitecture for integrated Information Systems) by using the UFO upper-level ontology. The objective is to present recommendations for improvements of the organizational language to resolve ontological miss-interpretation, semantic overload and construct redundancy.

In [22] the use of domain ontologies to extended an existing conceptual modeling language is described. Moreover, the presentation of an UML profile defined by a domain ontology that specifies a domain axiomatization in terms of concepts, relations between concepts and rules that govern these relations are described. Here, the goal is to improve the domain-specific quality of UML class diagrams.

On the other hand, several approaches describe the process to integrate ontologies resolving the interoperability problem among existing ontologies. In [23] a graphical interface called *ONION* is presented to integrate ontologies, while [24] proposes the use of a global ontology to integrate heterogeneous data sources.

Differently to the mentioned approaches, our proposal integrated an organizational model represented as ontology with a general or specific domain ontology, achieving a final representation in which each element of the model is related with one or more domain concepts through “is_a” links. The integration proposed at instance level with domain concepts permits not only to add additional, unambiguous information to the model, but also to improve the analysis process through the hierarchical structure of the general ontologies. We consider that a domain concept that integrate different model elements is a strong indicator for the need of a new business service to implement inside the organization.

7 Conclusion

In this paper, we presented an approach for integrating organizational visual models with general or domain ontologies. To support the integration of an annotated visual model with an ontology we developed the TAGOOn+ tool.

The integration process can be carried out on models represented in i^* , Tropos and Service-oriented i^* . This integration process is based on the identification of elements that are common between the analysed business model and the domain concepts represented in ontology, and on their relationships. In this manner, the semantic annotation is added to the business model. We consider that a domain concept that integrates different elements of the model could represent new business services to the organization. These new functionalities can be useful to delineate new business services, and to improve the understandability and expressiveness of a model, thus giving a necessary condition for model reuse.

Moreover, the semantic annotation helps to discover hidden relationships and the collaboration with domain experts, which could understand and improve the model with a reduced effort of alignment on the terminology. The illustrative example presented provides a preliminary evidence of the utility of our semantic annotation suggestions and the integration process of a visual model represented in the i^* variants with a general ontology. The integration permits to facilitate the reasoning between model elements and to add additional information clarifying the description of each element.

References

1. Yu, E.: Modelling strategic relationships for process reengineering. PhD thesis, University of Toronto, Toronto, Ont., Canada (1996)
2. Bresciani, P., Perini, A., Giorgini, P., Giunchiglia, F., Mylopoulos, J.: Tropos: An Agent-Oriented Software Development Methodology. *Autonomous Agents and Multi-Agent Systems* **8**(3) (2004) 203–236
3. Estrada., H.: A service-oriented approach for the i^* framework. PhD thesis, University of Technology, Valencia, Spain (2008)
4. Grau, G., Cares, C., Franch, X., Navarrete, F.J.: A Comparative Analysis of i^* Agent Oriented Modelling Techniques. In: *Proceedings of The Eighteenth International Conference on Software Engineering and Knowledge Engineering*. (2006)
5. Decreus, K., Poels, G.: Mapping semantically enriched Formal Tropos to business process models. In: *Proceedings of the ACM Symposium on Applied Computing. SAC '09*, New York, NY, USA, ACM (2009) 371–376
6. Koliadis, G., Vranesevic, A., Bhuiyan, M., Krishna, A., Ghose, A.: Combining i^* and BPMN for business process model lifecycle management. In: *Proceedings of the International Conference on Business Process Management Workshops. BPM'06*, Berlin, Heidelberg, Springer-Verlag (2006) 416–427
7. Deng, X.: Intentional Modeling for Enterprise Architecture - Managing Knowledge about "Why" to Support Change. Master's thesis, Faculty of Information Studies, University of Toronto (2006)
8. Gruber, T.R.: Toward Principles for the Design of Ontologies Used for Knowledge Sharing. In: *Formal Ontology In Conceptual Analysis and Knowledge Representation*, Kluwer Academic Publishers (1993)
9. Guarino, N.: *Formal ontology and information systems*, IOS Press (1998) 3–15
10. Nirenburg, S., Raskin, V.: *Ontological semantics*. MIT Press (2004)

11. Gangemi, A., Guarino, N., Masolo, C., Oltramari, A., Schneider, L.: Sweetening ontologies with dolce. In: Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web. EKAW '02, London, UK, Springer-Verlag (2002) 166–181
12. Guarino, N., Welty, C.: Evaluating ontological decisions with ontoclean. Commun. ACM **45**(2) (February 2002) 61–65
13. McShane, M., Beale, S., Nirenburg, S.: Ontosem methods for processing semantic ellipsis. In: Proceedings of the Workshop on Computational Lexical Semantics. HLT-NAACL 2004, Boston, Massachusetts, USA, Association for Computational Linguistics (May 2 - May 7 2004) 1–8
14. Nirenburg, S., Beale, S., McShane, M.: Baseline evaluation of WSD and semantic dependency in OntoSem. In: Proceedings of the Conference on Semantics in Text Processing. STEP '08, Stroudsburg, PA, USA, Association for Computational Linguistics (2008) 315–326
15. Vazquez, B., Martinez, A., Perini, A., Estrada, H., Morandini, M.: Enriching Organizational Model through Semantic Annotation. In: Proceedings of the Iberoamerican Conference on Electronics Engineering and Computer Science, Procedia Technology (April 24-26 2013)
16. Vazquez, B.: Enriching organizational models through semantic annotation. Master's thesis, National Center of Research and Technological Development, Cuernavaca, Morelos, Mexico (2012)
17. Cares, C., Franch, X., Perini, A., Susi, A.: iStarML: An XML-based Model Interchange Format for i*. In Castro, J., Franch, X., Perini, A., Yu, E., eds.: iStar Workshop. Volume 322 of CEUR Workshop Proceedings., CEUR-WS (2008) 13–16
18. Najera, K.: An ontology-based approach for integrating i* variants. Master's thesis, National Center of Research and Technological Development, Cuernavaca, Morelos, Mexico (2011)
19. Najera, K., Perini, A., Martínez, A., Estrada, H.: Supporting i* model integration through an ontology-based approach. In: iStar. (2011) 43–48
20. Yu, E.S.K., Liu, L.: Modelling trust for system design using the i* strategic actors framework. In: Proceedings of the workshop on Deception, Fraud, and Trust in Agent Societies held during the Autonomous Agents Conference: Trust in Cyber-societies, Integrating the Human and Artificial Perspectives, London, UK, UK, Springer-Verlag (2001) 175–194
21. Jr., P.S.S., Almeida, J.P.A., Guizzardi, G.: An ontology-based semantic foundation for organizational structure modeling in the aris method. In: Proceedings of the 14th IEEE International Enterprise Distributed Object Computing Conference Workshops, Washington, DC, USA, IEEE Computer Society (2010) 272–282
22. Gailly, F., Poels, G.: Conceptual modeling using domain ontologies: improving the domain-specific quality of conceptual schemas. In: Proceedings of the 10th Workshop on Domain-Specific Modeling. DSM '10, New York, NY, USA, ACM (2010) 18:1–18:6
23. Mitra, P., Wiederhold, G., Kersten, M.L.: A graph-oriented model for articulation of ontology interdependencies. In: Proceedings of the 7th International Conference on Extending Database Technology: Advances in Database Technology. EDBT '00, London, UK, UK, Springer-Verlag (2000) 86–100
24. Bergamaschi, S., Castano, S., Vincini, M.: Semantic integration of semistructured and structured data sources. SIGMOD Rec. **28**(1) (March 1999) 54–59